AD NUMBER

AD837562

LIMITATION CHANGES

TO:

Approved for public release; distribution is unlimited.

FROM:

**Distribution authorized to U.S. Gov't. agencies and their contractors;
Administrative/Operational Use; 31 MAY 1968.
Other requests shall be referred to Defense Atomic Support Agency, Washington, DC.**

AUTHORITY

NELC ltr 22 Aug 1974

A FORTRAN PROGRAM FOR MODE CONSTANTS IN AN EARTH-IONOSPHERE WAVEGUIDE

Interim Report No. 683

C. H. SHEDDY, R. PAPPERT, Y. GOUGH, and W. MOLER

NAVAL ELECTRONICS LABORATORY CENTER
SAN DIEGO, CALIFORNIA 92152

31 May 1968
Prepared for DEFENSE ATOMIC SUPPORT AGENCY
on DASA Subtask RHA 2042

NELC Problem M402

TABLE OF CONTENTS

FIGURES AND TABLES

ABSTRACT

An updated version of an earlier waveguide program has been written in the FORTRAN compiler language. The new program includes the following alterations or additions; (1) An improved Runge-Kutta routine; (2) A procedure which finds directly the starting solutions at the top of the ionosphere; (3) Provision for including up to five charged particles; (4) A method for economically finding approximate waveguide solutions; and modifications for use of the program for ELF wave propagation.

I. INTRODUCTION:

This report is the latest in a series (1, 2, 3, 4, 5, 6, 7) which describe computer programs which have been developed for calculating ionospheric reflection coefficients and/or waveguide modal parameters. All earlier computer programs were written in the NELIAC computer compiler language. This report describes a program written in FORTRAN 63 for the CDC computer which solves the modal equation for propagation within an earth ionosphere waveguide. The experienced programmer, familiar with the FORTRAN language, should find it easy to make the necessary modifications for use of the program in other computers.

The solutions of the modal equation are determined from the reflection coefficient matrixes of the ionosphere and of the earth as seen looking upward and downward from a position within the waveguide. The program systematically solves the modal equation for as many modes as are desired. The propagation parameters, attenuation rate, phase velocity, and excitation factor are determined for each solution of the modal equation. The method makes full allowance for both earth curvature in the direction of propagation and the orientation and intensity of the geomagnetic field with respect to the path of propagation.

The reflection coefficient matrix of the ionosphere is obtained numerically following methods developed by Budden (8). The ionosphere is described in terms of the vertical distribution of charged particle density, the appropriate charged particle neutral particle collision frequencies as a function of height, and the intensity, dip angle, and azimuth of the geomagnetic field. As many as five charged particles of either sign and of arbitrary mass may be included as constituents. At the option of the user the program may be used to calculate and print a listing of the ionospheric reflection coefficients only.

In addition to the real constituents the medium contains a fictitious linear distribution of permittivity extending to the ground which simulates (takes into account) the effects of earth curvature.

The equations used for obtaining the reflection coefficient matrix looking at the ground depend on the height at which the modal equation is solved. If the modal equation is evaluated below the ionosphere but above the earth, because of the linear distribution of permittivity, the surface reflection coefficients are expressible in terms of solutions to Stokes' equation and

their derivatives. If the modal equation is evaluated at the ground all earth curvature effects are included in the ionospheric reflection coefficients and ordinary Fresnel reflection coefficients are obtained at the ground. The ground conductivity is an arbitrary input so that results represent propagation over a realistic finitely conducting earth.

The modal equation is especially useful for calculating the effect on elf, vlf, and lf guided propagation by anomalous conditions such as caused by solar flares or nuclear explosions. Step models or exponential models of the ionosphere are often so unrealistic as to be virtually useless for evaluating the effects of such anomalies. The present program calculates the attenuation, phase velocity, and excitation of important waveguide modes for realistic waveguide parameters. The program thus allows one to make realistic estimates of changes in field strength and phase velocity over critical paths.

In summary, the physical input parameters for the waveguide program are:

1. The azimuth of propagation with respect to the horizontal component of the geomagnetic field.

2. The geomagnetic dip angle and field strength.

3. The radio frequencies.

4. Vertical profiles of up to five charged species (electron, positive and negative ions).

5. An exponential but otherwise arbitrary collision frequency distribution with height for each specie of charged particles.

6. The ground conductivity and dielectric constant.

The output can be either a printout of the reflection coefficient matrix of the ionosphere at desired height increments within and below the ionosphere or, by using the complete program, the waveguide mode parameters for as many modes and radio frequencies as desired.

II. THE IONOSPHERE REFLECTION MATRIX

A. Exact Solution

The ionosphere reflection matrix, $\underset{\sim}{R}$, at height, d, is obtained by a numerical integration of differential equations given by Budden [8]. The differential equations are of the form

$$2i\underset{\sim}{R}' = \underset{\sim}{S}^{(21)} + \underset{\sim}{S}^{(22)}\underset{\sim}{R} - \underset{\sim}{R}S^{(11)} - \underset{\sim}{R}\underset{\sim}{S}^{(12)}\underset{\sim}{R} \tag{II-1}$$

where $$\underset{\sim}{S} = \left(\begin{array}{c|c} S^{(11)} & S^{(12)} \\ \hline S^{(21)} & S^{(22)} \end{array}\right)$$

Because of some symmetry in the elements of the S-matrix, it may be written in the form

$$\underset{\sim}{S} = \left(\begin{array}{cc|cc} S_{11}a + S_{11}b & -S_{12} & -d_{11}a + d_{11}b & -S_{12} \\ -S_{21} & S_{22} & -d_{12} & -d_{22} \\ \hline -d_{11}a - d_{11}b & d_{12} & S_{11}a - S_{11}b & d_{12} \\ S_{21} & d_{22} & d_{21} & -S_{22} \end{array}\right) \tag{II-2}$$

where

$$\begin{aligned} S_{11}a &= T_{11} + T_{44} \\ d_{11}a &= T_{11} - T_{44} \\ S_{11}b &= T_{14}/C - CT_{41} \\ d_{11}b &= T_{14}/C - CT_{41} \\ S_{12} &= T_{12}/C + T_{42} \\ d_{12} &= T_{12}/C - T_{42} \\ S_{21} &= T_{34}/C + T_{31} \\ d_{21} &= T_{34}/C - T_{31} \\ S_{22} &= C + T_{32}/C \\ d_{22} &= C - T_{32}/C \end{aligned}$$

where C = cos θ and θ is the complex angle of incidence. The T-matrix is of the form

$$\underset{\sim}{T} = \left(\begin{array}{c|c|c|c} -Sm_{31}D & Sm_{32}D & 0 & (C^2+m_{33})D \\ 0 & 0 & 1 & 0 \\ m_{23}m_{31}D-m_{21} & C^2+m_{22}-m_{23}m_{32}D & 0 & Sm_{23}D \\ 1+m_{11}-m_{13}m_{31}D & m_{32}m_{13}D-m_{12} & 0 & -Sm_{13}D \end{array}\right) \qquad \text{(II-3)}$$

where

$$D = 1 + m_{33}$$
$$S = \sin\theta \qquad \text{(II-4)}$$

and

$$\underset{\sim}{m} = \frac{-X}{U(U^2-Y^2)} \left(\begin{array}{c|c|c} U^2 - Y_x^2 & -iY_zU - Y_xY_y & iY_yU - Y_xY_z \\ iY_zU - Y_xY_y & U^2 - Y_y^2 & -iY_xU - Y_yY_z \\ -iY_yU - Y_xY_z & iY_xU - Y_yY_z & U^2 - Y_z^2 \end{array}\right)$$

$$X = \frac{N_e^2}{\epsilon_0 m \omega^2}$$

$$\vec{Y} = \frac{\mu_0 e \vec{H}_e}{m\omega} \qquad \vec{Y} \text{ and } \vec{H}_e \text{ in opposite direction for -e)}$$

$$Z = \frac{\nu}{\omega}$$

$$U = 1 - iZ$$

ϵ_0 = electric permittivity of free space

μ_0 = magnetic permeability of free space

$\vec{H}_e$ = geomagnetic field

ω = angular wave frequency

The quantities N, e, m, ν (nu) have different values, at a given height, for each species. The quantities are

N = species density
e = charge of electron or of ion (negative for electrons and negative ions)
m = mass of electron or of ion
ν = collision frequency

For each species there will be an $\underset{\sim}{M}_K$. The matrix, $\underset{\sim}{M}$, used in computing the T-matrix is then

$$\underset{\sim}{M} = \sum_k \underset{\sim}{M}_k \qquad \text{(II-5)}$$

The differential equations are integrated by the Runge-Kutta technique, starting at some height above which negligeble reflection is assumed to take place. Error control is by means of a comparison, for each step, of the increments in the elements of $\underset{\sim}{R}$ computed with the fourth order Runge-Kutta method, and that computed with a second-order integration step. Although not a completely rigorous error test, it has been found to be very reliable.

The eight integration variables are the arguments and the logs of the magnitudes of each of the four elements of the reflection matrix. The derivatives of these variables are obtained with the expression

$$(\log R_{ij})' = \frac{R'_{ij}}{R_{ij}} \qquad \text{(II-6)}$$

where $\log R_{ij}$ implies the complex logarithm of the ij^{th} element of $\underset{\sim}{R}$.

It turns out that the initial conditions for the integration, i.e., the starting value of $\underset{\sim}{R}$, is the value of $\underset{\sim}{R}$ for a sharply-bounded ionosphere with properties corresponding to the ionosphere at the top of the given electron density and collision frequency profiles. The solution used is that described in a paper by C.H. Sheddy, (9).

B. Approximate Solution

Most of the running time of the NELC waveguide program is in the computation of $\underset{\sim}{R}$ which is used in the modal equation $\left|\underset{\sim}{R}\underset{\sim}{R}_g - 1\right| = 0$. Since the eigen values of θ usually occur along a line in the complex θ-plane, it often suffices to compute $\underset{\sim}{R}$ for two or three values of θ and interpolate between these values to obtain $\underset{\sim}{R} = f(\theta)$. In this way the total run time is cut to a fraction of that for the exact solution.

Approximate solutions may be used for exploratory work where one does not require great accuracy. It may also be used in cases where the electron density profile itself is not known too well, since the errors resulting from use of the interpolation may be no greater than the uncertainties due to not knowing the profile too well. Even when exact solutions are desired, approximate solutions may be obtained first as starting values for exact solutions.

The Lagrange interpolation polynomial is used in the program:

$$\log R(\theta) = \sum_{k=1}^{n} \prod_{\substack{j=1 \\ j\neq k}}^{n} \frac{\theta-\theta_j}{\theta_k-\theta_j} \log R_k \qquad \text{(II-7)}$$

where log R implies the complex log of an element of $\underset{\sim}{R}$. R_k is a value computed with the full-wave solution at the angle, θ_k.

The user must select n values of θ_k when using the approximate scheme. Three or four values of θ_k generally give best results, often two suffice. For best results these should be chosen as being in a line in the complex θ-plane roughly approximating the line along which the eigen solutions are expected to be. It also has seemed best not to have more than one θ_k with the same value for the real parts or the same value for the imaginary parts.

When using the waveguide program for the first time, the user might try several sets of θ_k for the same set of input data in order to gain a feel for the sensitivity of the approximate eigen solutions on the θ_k.

III. SUMMARY OF MODE PARAMETERS

Following the formalism developed by Budden, [10] whereby one works within the framework of a planar stratified medium and includes earth curvature via the artifice of a modified refractive index, it is well known that the mode equation may be written as the determinantal equation,

$$F_D(\theta) = \left| \overline{R}_D(\theta) R_D(\theta) - 1 \right| = 0 \tag{III-1}$$

where

$$R_D(\theta) = \begin{pmatrix} {}_{\parallel}R_{\parallel D}(\theta) & {}_{\perp}R_{\parallel D}(\theta) \\ {}_{\parallel}R_{\perp D}(\theta) & {}_{\perp}R_{\perp D}(\theta) \end{pmatrix} \tag{III-2}$$

is the reflection matrix looking up into the ionosphere from height D and

$$\overline{R}_D(\theta) = \begin{pmatrix} {}_{\parallel}\overline{R}_{\parallel D}(\theta) & 0 \\ 0 & {}_{\perp}\overline{R}_{\perp D}(\theta) \end{pmatrix} \tag{III-3}$$

is the reflection matrix looking down from height D towards the ground. The angle, θ, is the angle of incidence at height z = H where the modified index of refraction

$$n^2 = 1 - \alpha(H - Z) \tag{III-4}$$

is unity. The full wave equations programed for the purpose of determining the elements of $R_D(\theta)$ are given in section II. It is emphasized that earth curvature is included in the calculations of the elements $R_D(\theta)$, by the susceptibility matrix. Since $\overline{R}_D(\theta)$ is assumed to be diagonal it is essential that the level D be chosen at or below the base of the ionosphere. Except for D = 0 and

some ELF calculations, $\overline{R}_D$, is calculated as described in reference [11] in terms of solutions to Stokes' equation and their derivatives. The solution to Stokes' equations which we have chosen to work with are the modified Hankel functions of order 1/3 as defined in reference [12]. It should be observed that the latter functions are linearly related to the perhaps better known Airy functions. When D = 0 or when the magnitude of the hyperbolic sine of the imaginary part of the eigen angle exceeds a preassigned limit, the $\overline{R}_D(\Theta)$ matrix is calculated directly from the Fresnel reflection formulas.

The principal objective of the present work is the solution of (1) for the eigen angles, θ_n. To achieve this the well known Newton's method is used. The procedure is to guess a solution θ_o to the equation $F_D(\Theta)=0$. The function $F_D(\Theta)$ is then reevaluated for $\theta_o + \delta\theta$ and the correction to θ_o found from the equation

$$\Delta\Theta = - \frac{F(\Theta)\,\delta\Theta}{F(\Theta_o + \delta\Theta) - F(\Theta_o)} \qquad \text{(III-5)}$$

The correction determined by Eq. (5) is then evaluated and the process repeated until the quantities $|\Delta\Theta_r|$ and $|\Delta\Theta_i|$ are reduced to within a preassigned tolerance. For the first iteration the value of $\delta\theta$ is taken equal to 0.01°. The subsequent choices of $\delta\theta$ are contigent upon proximity of the results of the last two iterations.

With the eigen angle, θ_n, for mode n determined the following mode parameters are readily evaluated

$$\text{phase velocity at the ground} = \frac{C}{K(\sin\Theta_n)_r}\left(\frac{km}{sec}\right) \qquad \text{(III-6)}$$

$$\text{attenuation constant at the ground} = -8.6859\,kK(\sin\Theta_n)_i\left(\frac{db}{Mm}\right) \qquad \text{(III-7)}$$

excitation factor as defined by Wait[4] = (III-8)

$$-i\frac{kh'}{2}\sin\Theta_n \quad \frac{(1+{}_{\parallel}\overline{R}_{\parallel})^2(1-{}_{\perp}\overline{R}_{\perp}R_{\perp})}{{}_{\parallel}\overline{R}_{\parallel}\left(\frac{\partial F}{\partial\Theta}\right)_{\Theta=\Theta_n}} f^2$$

Mode mixing parameter (printed out as polarization) = (III-9)

$$-\frac{{}_{\parallel}\overline{R}_{\parallel\perp}R_{\parallel}}{({}_{\parallel}\overline{R}_{\parallel\parallel}R_{\parallel}-1)}$$

where $K = (1+\alpha H/2)$ (III-10)

$$f = \exp\left(-\frac{\alpha D}{2}\right)\left[\frac{F_1 h_1(q_0)+F_2 h_2(q_0)}{F_1 h_1(q_D)+F_2 h_2(q_D)}\right] \quad \text{(III-11)}$$

$$F_1 = -\left[h_2'(q_0)+\frac{1}{2}\left(\frac{\alpha}{k}\right)^{2/3}h_2(q_0)-i\frac{n_0^2}{N_g^2}\left(\frac{k}{\alpha}\right)^{1/3}\left(N_g^2-\sin^2\Theta_n\right)^{1/2}h_2(q_0)\right] \quad \text{(III-12)}$$

$$F_2 = \left[h_1'(q_0)+\frac{1}{2}\left(\frac{\alpha}{k}\right)^{2/3}h_1(q_0)-i\frac{n_0^2}{N_g^2}\left(\frac{k}{\alpha}\right)^{1/3}\left(N_g^2-\sin^2\Theta_n\right)^{1/2}h_1(q_0)\right] \quad \text{(III-13)}$$

$$N_g^2 = \frac{\varepsilon}{\varepsilon_0}-i\frac{\sigma}{\omega\varepsilon_0} =$$ square of the complex index of refraction of the ground (III-14)

$$q = \left(\frac{k}{\alpha}\right)^{2/3}\left(\cos^2\Theta_n-\alpha(H-Z)\right) \quad \text{(III-15)}$$

The subscript r in Eq. (6) implies the real part while the subscript i in Eq. (7) implies the imaginary part. The subscript on q represents the value of z at which q is evaluated. For example, q_o means that Eq. (15) is to be evaluated for Z =0. Similarly, the subscript in n^2 represents the value of Z for which Eq. (4) is to be evaluated. The functions h_1 and h_2 are the modified Hankel functions of order 1/3. The primes in Eqs. (12) and (13) denote derivatives with respect to the argument. The quantity h' which occurs in Eq. (8) is called REFL HT in the program and has been introduced solely for the purpose of normalizing the excitation factor for the vertical dipole in a way which is consistent with Wait's definition [13]. Closely related quantities also included in the print out are termed EXTRA MAG and EXTRA ANGLE. These quantities are simply the magnitude and argument of the right hand side of Eq. (8) with the factor $(-i\frac{kh'}{2})$ divided out. The quantities EXTRA MAG and EXTRA ANGLE are used in conjunction with the mode sum analysis discussed in section IV. The quantity printed out as polarization, that is the right hand side of Eq. (9), is in fact not a measure of the polarization but as shown in reference 14 is very closely related to the ratio of the vertical to horizontal components of polarization which comprise each mode under the general condition of an anisotropic ionosphere. The right hand side of Eq. (9) becomes meaningless under conditions of an isotropic or near isotropic ionosphere. For then the numerator of equation (9) is very small because of a small conversion coefficient. On the other hand the denominator of Eq. (9) even for a vertically polarized mode will not vanish in general because of round off error so that the entire ratio is then dependent upon round off error and in fact can have magnitude less than unity even through the mode is vertically polarized.

Finally, the quantity printed out under the label THETA PRIME, is simply the value of the eigen angle at the ground (recall that the eigen angle which results from the process of interation is referred to height H).

IV. CALCULATION OF RADIO FIELD STRENGTH

For the frequencies considered for waveguide propagation the transmitting antenna is generally a vertical monopole above the surface of the earth. Because the earth is a good conductor the radiation characteristic of the antenna is virtually that of a dipole radiating into a half space. The power radiated Pr is then

$$P_r = 160\pi^2\left(\frac{l}{\lambda}\right)^2 I^2 \text{ watts} \qquad \text{(IV-1)}$$

where I is the rms current, l is the effective height of the antenna and λ is the radio wavelength. The electric field at a distance D from the transmitter for the n th mode can be written as:

$$E_n = \left[\frac{e^{i\frac{\pi}{4}}\eta}{4\pi}\left(\frac{C}{fa}\right)^{1/2}\right]\left(\frac{P_r}{f}\right)^{1/2}\frac{\exp\left[-ikD(1-C_n^2/2)\right]}{(a\sin D/a)^{1/2}}\frac{kX}{2}\sin^{3/2}\theta_n \text{ V/m} \qquad \text{(IV-2)}$$

where η is 377 ohms, c is the speed of light, f is radio frequency, a is the radius of the earth, C_n is the cosine of θ_n the n th eigen angle, k is wave number, and x in the magnitude of the exitation factor (extra mag.). A practical form to use is

$$E(v/m/kw) = C_1 e^{-C_2\alpha} X (\sin\theta)^{3/2}$$

where

$$C_1 = \frac{.2041}{\lambda}\left(f\sin\frac{D}{a}\right)^{-1/2} \qquad \text{(IV-3)}$$

$$C_2 = .1152\,D$$

and α = attenuation coefficient in db/megameter.

The relative phase of the signal (relative with respect to free space propagation or to a local oscillator) is given by

$$\varphi = \frac{2\pi f}{c} D\left(1 - \frac{c}{v}\right) + \varphi_o \qquad \text{(IV-4)}$$

where φ is the relative phase at the receiver, c is the speed of light, v is the phase velocity and φ_o is the argument of the excitation factor in radians.

V. RUNNING THE PROGRAM

A. The profile deck(s)

The program, whether run for WAVEGUIDE SOLUTIONS or for REFLECTION COEFFICIENTS can be run with or without ion density profiles. The profiles are set up the same way for either case. The electron density profile comes first and is preceded by a SPECIES 1 PROFILE card and an alpha-numeric card punched with information needed to identify the run. The profile starts at the top of the ionosphere and contains the height in kilometers and the electron density/cubic centimeter for that height. The height is punched in columns 2-7 and the density is punched in columns 14-21. The decimals are in columns 5 and 15. If an electron density of zero is used the program will change it to 10^{-40}. Although the program is often run with an electron density for every half kilometer, it may be run using only the significant heights. For example, only two cards are needed for an exponential profile: One at the top and one at the bottom of the profile. The program interpolates in a straight line between the logs of the densities and heights given in the profile deck. The profile deck ends with a card punched with nines in the first 8 columns. If the program is to be run without ions, the input parameters immediately follow this nines card. If the program is to be run with ions, this nines card is followed by a SPECIES 2 PROFILE card, another card with identification, then the top of the positive ion density profile and so on. This also ends with a nines card followed by a SPECIES 3 PROFILE card, an identification card, and the negative ion density profile. Again a nines, and then the input parameters. However, since the negative ion density is equal to the positive ion density less the electron density for each height, the program will calculate the negative ion density if a NEGIONS card is placed at the end of the positive ion density profile. The ion density profiles must have the same heights as the electron density profile. A sample profile with ions included is shown in Table A.

B. Waveguide Solution Input Parameters

The input parameters are placed anywhere on the card but according to the following format. The names, e.g., MAGFIELD, must have no spaces between the letters. The name is followed by one space, an = sign, another space, and then the value of the parameter. If a parameter has more than one value they are listed on the card or on following cards with one or more spaces between them. The parameter name is only used once.

Table A

```
SPECIES 1 PROFILE
SAMPLE PROFILE ELECTRON DENSITY
 100.00        8.18E+04
  95.00        6.02E+04
  90.00        8.92E+04
  85.00        1.26E+05
  80.00        1.75E+05
  75.00        2.42E+05
  70.00        2.86E+05
  65.00        2.65E+05
  60.00        1.73E+05
  55.00        1.88E+04
  50.10        3.85E+03
  45.00        1.03E+03
  40.00        9.86E+02
  35.00        4.66E+02
  30.00        1.49E+02
  25.00        2.84E+01
  20.00        2.02E+00
  15.00        2.00E-02
  10.00        3.17E-05
   5.00        5.87E-06
   0.          3.07E-06
99999999
SPECIES 2 PROFILE
SAMPLE PROFILE POSITIVE ION DENSITY
 100.00        8.18E+04
  95.00        6.03E+04
  90.00        8.93E+04
  85.10        1.27E+05
  80.00        1.78E+05
  75.00        2.65E+05
  70.10        4.26E+05
  65.00        7.89E+05
  60.00        1.41E+06
  55.00        2.25E+06
  50.00        2.69E+06
  45.00        3.10E+06
  40.00        3.49E+06
  35.00        3.76E+06
  30.00        3.33E+06
  25.00        2.22E+06
  20.00        8.78E+05
  15.00        1.25E+05
  10.00        6.16E+03
   5.00        4.47E+03
   0.00        5.49E+03
99999999
REGIONS
```

The approximate method will find up to 25 solutions in 10 minutes and the exact method might find one solution in that amount of time. Therefore it expedites the search for mode solutions. Often these solutions are sufficiently accurate for the purpose. If this is not the case they can serve as excellent starting solutions when running the exact method. To set up an approximate run, only two cards must be added to an exact waveguide run. These are the RPOLY card and the TLIST card which are explained below.

AZIMUTH is the direction of propagation in degrees, measured east of magnetic north. West-to-east propagation corresponds to an azimuth of 90° and east-to-west propagation corresponds to an azimuth of 270°.

DIPANGLE is the magnetic dip angle in degrees, measured from the vertical and is always between 0 and 90 degrees for propagation in the northern himisphere and between 90 and 180 degrees for propagation in the southern hemisphere.

MAGFIELD is the intensity of the magnetic field in Webers/square meter. Use 0.0 for no mag field.

COEFFNU and EXPNU are the values used to find ν, the collision frequency where

$$\nu = \text{COEFFNU} \times \exp(\text{EXPNU} \times Z)$$

and Z is in meters, EXPNU is in inverse meters and COEFFNU is in collisions/second. If ions are included, a separate collision frequency is needed for each species and is listed on each card in the order of the profiles. For example COEFFNU = 4.303E11 1.076E10 1.076E10 would mean that the coefficient of the collision frequency is 4.303E11 (4.303×10^{11}) for the electron density profile and 1.076E10 for the two ion density profiles.

EPSILON and EPSILON0 are the permittivities of the ground and free space respectively and are to be expressed in units of farads/meter.

D is any height below which ionospheric effects can be considered negligible and must be consistent with the bottom of the profile deck. Care must be taken in satisfying the preceding condition. If there is doubt as to whether a suitable value has been chosen, it is advisable to run the program to different heights, D, to verify that the eigen-angles have stabilized. Generally, stability to better than $\pm$ 0.05° in the ground eigenangle is obtained.

H is the height, expressed in kilometers, which represents physically the height at which the modified refractive index becomes unity and is the height at which the unprimed eigenangle is measured. Theory, good to first order in the parameters h/a and z/a, predicts that the final ground solutions are independent of H. However, investigation has shown that the results vary somewhat when H changes from 90 km to the ground (by approximately a tenth of a degree in the ground eigen angle). This variation is attributed to higher order effects in the earth curvature correction. Although contributing to some uncertainty in the final results, these variations can be considered small and are sufficiently accurate for most purposes. Generally, it is sufficient to choose both D and H to be the height on the last card of the electron density profile.

ALPHA is in inverse kilometers and is equal to 2/(earth's radius in kilometers). If flat earth programs are run, a very small number (e.g. 10^{-6} km^{-1}) should be used instead of zero.

RELPREC is a parameter which helps determine how accurately the calculations are to be done. A RELPREC of 1.0 would be fast but not very accurate, 2.0 is usual, and 3.0 or 4.0 would give about maximum accuracy.

SIGMA is the ground conductivity expressed in units of mhos/meter.

REFLHT is chosen to be a reasonable estimate of where reflection takes place. It is used in both the approximate and the exact methods as a factor in obtaining WAITS EXCITATION FACTOR. It is also used in obtaining the approximate solutions. It is in kilometers.

FREQKHZ is the radio frequency in kilohertz/second.

THETAINC is a card used to control the iterations. A THETAINC of 1.0 insures that each succeeding iteration will be no more than one degree (in real and in imaginary parts) from the preceeding one. This keeps the program from arriving at a solution other than the one desired. However, if little is known about the eigen angle solutions, a THETAINC of 5.0 or even greater would be best.

MRATIO is used only when ions are included. The first number is unity; the second the ratio of positive ion mass to electron mass; and the third, the ratio of negative ion mass to electron mass.

CHARGE is also necessary only when ions are included. The first number is the charge of electrons, -1.0, the second, the charge of positive ions, + 1.0, and the third the charge of negative ions, -1.0.

RPOLY = 1 if approximate solutions are desired.

TLIST is a list of complex angles used in the approximate method to obtain exact reflection coefficients which are used in an interpolation scheme to find the eigenangles of the problem. The closer the elements of the TLIST are to the final eigen angle solutions the closer the solutions will be to exact solutions. For example, if all modes were derived between 70° and 80°, the TLIST might look like the following:

TLIST = 70 -3 75 -2 80 -1

Then all eigens in that range could be run in very little computer time. E.g. the EIGEN card might appear as

EIGEN = 70.6 -2.7 73.5 -1.2 74 -1 78 -2.5

If that is not accurate enough, and an approximate solution near (76, -1.6)° is desired the TLIST might look like the following:

TLIST = 75 -1.8 76 -1.6 77 -1.4

EIGEN precedes the trial eigen angles. As many eigen angles as desired are punched on cards in the format of real part, imaginary part, real part, imaginary part, and so on. Each number separated by one or more spaces. If more than one card is needed the numbers are just continued on the following cards without reusing the same EIGEN.

WAVEGUID is the command to the computer to start and complete the waveguide solutions.

See Table B for a sample approximate waveguide run with ions included. These parameters would follow the profile decks. For the run to be the exact method, only the RPOLY card and the TLIST card need be removed.

C. Reflection Coefficient Input Parameters

FREQKHZ	(See Waveguide input parameters)
AZIMUTH	"
DIPANGLE	"
MAGFIELD	"
COEFFNU	"
EXPNU	"
D	"
H	"
ALPHA	"
RELPREC	"

Table B

```
AZIMUTH = 83.0
DIPANGLE = 17.0
MAGFIELD = 5.0E-05
COEFFNU = 4.303E11  1.076E10  1.076E10
EXPNU = -1.622E-04  -1.622E-04  -1.622E-04
EPSILON = 7.1720154E-10
EPSILONO = 8.85434E-12
D = 0.0
H = 0.0
ALPHA = 3.14E-04
RELPREC = 2.0
SIGMA = 4.64
REFLHT = 60.0
MRATIO = 1.0  58000.0  56000.0
CHARGE = -1.0  1.0  -1.0
THETAINC = 1.0
RPOLY = 1
TLIST = 75.5 -3.0 82.0 -2.0 88.5 -1.0
EIGEN = 75.0 -2.0 76.0 -2.0 77.0 -2.0 78.0 -2.0 79.0 -2.0 80.0 -2.0
82.0 -2.0 83.0 -2.0 84.0 -2.0 85.0 -2.0 86.0 -2.0 87.0 -2.0 88.0 -2
87.5 -2.0 88.5 -2.0 89.5 -2.0
FREQKHZ = 15.0
WAVEGUID
```

MRATIO (if ions are in- "
cluded)

CHARGE " "

THETA is the complex angle of incidence in degrees.

See Table C for a sample input for reflection coefficients for electrons only. The reflection coefficient program only runs exact.

RC is the command to the computer to start and complete the reflection coefficients.

D. Linking Several Computer Runs

By doing several different runs in one computer input one can sometimes save on turnaround time and cut down on card duplication of either inputs or profiles.

More than one data deck is run at a time by following the WAVEGUID card with a SPECIES 1 PROFILE card and so on through the profile or through the ion profiles as before. Doing this clobbers the profiles which were originally in the computer but all the other input remains as it was. At the next WAVEGUID card the program starts on this "second" program. If any changes in input parameters were desired on this second run, they would be changed between the profile and the WAVEGUID card. For example, if D = 0.0 on the first profile and D = 65.0 on the second, the new profile would end with a nines card and then NEGIONS, if ions were included, then D = 65.0 and then WAVEGUID.

Changes in the parameters for the data deck are stated between WAVEGUID cards. All other inputs would remain unchanged. For example, if several mag fields are desired for a given profile and set of parameters, only mag field would be changed before calling WAVEGUID again. For example

MAGFIELD = 4.0E-05
WAVEGUID
MAGFIELD = 0.0
WAVEGUID

It is sometimes justifiable under disturbed conditions such as SID's or man-made ionization anomalies to neglect the influence of the earth's magnetic field. The modal equation then factors into a product, one factor of which contains solutions for vertical polarization, while the other factor contains solutions for horizontal polarization. A provision is thus made whereby under those circumstances either the vertical or the horizontal solutions may be obtained independently. Of course for vertical dipole excitation, the vertically polarized modes are the physically meaningful modes. So when vertical modes are desired and the profile is run with no mag field, the command TYPEITER= 1

Table C

```
SPECIES 1 PROFILE
SAMPLE PROFILE ELECTRONS ONLY
 100.00        7.02E 04
  95.00        3.06E 04
  90.00        1.33E 04
  85.00        6.12E 03
  80.00        2.80E 03
  75.00        1.34E 03
  70.00        5.72E 02
  65.00        2.07E 02
  60.00        8.40E 01
  55.00        6.50E 00
  50.00        4.27E-01
  45.00        1.60E-01
  40.00        5.42E-02
  35.00        1.58E-02
  30.00        4.03E-03
  25.00        9.54E-04
  20.00        2.19E-04
  15.00        5.32E-05
  10.00        1.47E-05
   5.00        5.88E-06
   0.00        0.00E 00
99999999
AZIMUTH = 33.0
DIPANGLE = 17.0
MAGFIELD = 5.0E-05
EXPNU = -1.622E-04
COEFFNU = 4.303E11
D = 0.0
H = 0.0
ALPHA = 3.14E-04
RELPREC = 2.0
FREQKHZ = 15.0
THETA = 88.5 -1.5
RC
```

will force only parallel solutions. If desired, horizontal solutions only will result from the command TYPEITER = 2. Either command may be used when propagation E-W or W-E is run at the geomagnetic equator (i.e. propagation direction transverse to the magnetic field).

E. Waveguide Solution Computer Printout

Line	Explanation of Table D
1, 2	Real and imaginary parts of the eigen angle followed by their reflection coefficients.
3	Difference between final two eigen angles.
4	Real and imaginary parts of the F function. See III.
5	Phase velocity of mode at the earth's surface.
6	Phase velocity divided by the speed of light.
7	Attenuation of the vertical E-field in dB/1000 km at the ground.
8, 9	Magnitude and angle of Wait's excitation factor which has been normalized by dividing out the -ikh'/2 term.
10, 11	Wait's excitation factor explained in (8). These excitation factors are for a vertical dipole antenna. See III-8.
12, 13	These give an indication if the eigen angle is horizontally or vertically polarized and are explained in III-9.
14	The eigen angle at the ground.

F. Reflection Coefficient Computer Printout

Explanation of Table E.

The computer prints out the height first then across the page, the modulus and argument in radians of ${}_{\parallel}R_{\parallel}$, ${}_{\perp}R_{\perp}$, ${}_{\perp}R_{\parallel}$, ${}_{\parallel}R_{\perp}$. At the bottom of the printout it prints ${}_{\perp}R_{\parallel}$ / ${}_{\perp}R_{\perp}$ and ${}_{\perp}\theta_{\parallel} - {}_{\perp}\theta_{\perp}$.

ITERATION TYPE 0
MAX DELTA = 0

	THETA	11R11 BAR	1R1 BAR	11R11	1R1	1R11	11R1	R11 BAR*R11	R1 BAR*R1
REAL	82.000	9.95443E-01	-9.99922E-01	6.19770E-01	6.72562E-01	-3.83998E-05	-3.38321E-05	6.17855E-01	-6.72521E-01
IMAG	-.500	-4.30778E-03	8.86474E-05	2.26728E-01	1.30428E-01	4.44513E-05	4.44848E-05	2.23210E-01	-1.30358E-01
MAG		.99545	.99992	.65994	.68509	.00006	.00006	.65694	.68504
ANGLE		359.76932	179.99492	20.09379	10.97494	130.82253	127.25412	19.86311	190.96986

THETA = 82.000 -.500 F MAG = 7.424E-01 DFDTHETA MAG = 3.066E-01

THETA = 81.000 -1.282 F MAG = 7.424E-01 DFDTHETA MAG = 3.066E-01

THETA = 81.355 -2.057 F MAG = 3.746E-01 DFDTHETA MAG = 4.393E-01

THETA = 81.253 -1.911 F MAG = 1.421E-01 DFDTHETA MAG = 7.966E-01

THETA = 81.246 -1.902 F MAG = 7.981E-03 DFDTHETA MAG = 7.080E-01

THETA = 81.246 -1.902 F MAG = 1.309E-04 DFDTHETA MAG = 7.031E-01

	THETA	11R11 BAR	1R1 BAR	11R11	1R1	1R11	11R1	R11 BAR*R11	R1 BAR*R1
REAL	81.246	9.95431E-01	-9.99928E-01	1.00462E 00	1.05932E 00	-5.49733E-05	-4.70412E-05	1.00004E 00	-1.05923E 00
IMAG	-1.902	-2.93883E-03	1.11010E-04	2.91683E-03	-1.74541E-01	8.68807E-05	8.57198E-05	-4.89048E-05	1.74646E-01
MAG		.99544	.99993	1.00463	1.07360	.00010	.00010	1.00004	1.07353
ANGLE		359.83085	179.99364	.16635	350.64364	122.32339	118.75702	359.99720	170.63728

ITERATIONS PERFORMED = 5

DELTHETA = .0001149 .0001464

F FUNCTION = -.00007 .00011

PHASE VELOCITY = 3.03159E 05 KM PER SEC
PHASE VELOCITY OVER C = 1.01123

ATTENUATION = 1.38010E 01 DB

EXTRA MAG = 2.12940E-01
EXTRA ANGLE = 1.76298 RAD

PHASE OF WAITS EXCITATION FACTOR = .21218 RAD
WAITS EXCITATION FACTOR IN DB = 5.93877E 00 DB

POLARIZATION MAG = 1.61625E 00
POLARIZATION ANGLE = 352.71763 DEG

THETA PRIME = 81.246 -1.902 DEG

Table D

R-MATRIX INTEGRATION

HT	11R11		1R1		1R11		11R1	
100.00	.0756	189.24	1.0000	179.73	.0389	43.84	.0388	43.71
99.97	.0758	189.39	1.0000	179.73	.0389	43.83	.0388	43.70
99.91	.0777	189.93	1.0000	179.73	.0389	43.81	.0388	43.68
99.78	.0849	190.19	1.0000	179.72	.0388	44.05	.0387	43.91
99.66	.0922	188.30	1.0000	179.72	.0389	44.48	.0389	44.34
99.53	.0962	186.28	1.0000	179.72	.0394	44.57	.0393	44.43
99.41	.0986	186.13	1.0000	179.71	.0397	44.23	.0396	44.10
99.28	.1032	187.35	1.0000	179.71	.0397	43.95	.0396	43.81
99.14	.1107	187.76	1.0000	179.71	.0396	44.07	.0395	43.93
99.03	.1179	186.67	1.0000	179.70	.0398	44.42	.0397	44.27
98.78	.1254	185.62	1.0000	179.70	.0404	44.17	.0403	44.02
98.53	.1360	185.66	1.0000	179.69	.0406	44.22	.0405	44.07
98.29	.1470	185.03	1.0000	179.68	.0410	44.22	.0409	44.07
98.03	.1573	184.91	1.0000	179.67	.0413	44.17	.0412	44.01
97.79	.1679	184.60	1.0000	179.67	.0417	44.19	.0416	44.03
97.53	.1781	184.39	1.0000	179.66	.0420	44.15	.0419	43.99
97.28	.1884	184.22	1.0000	179.65	.0423	44.14	.0422	43.97
96.78	.2098	183.54	1.0000	179.63	.0429	44.22	.0427	44.05
96.53	.2190	183.53	1.0000	179.62	.0433	44.13	.0432	43.95
96.29	.2290	183.64	1.0000	179.61	.0436	44.03	.0435	43.85
				•				
				•				
				•				
				•				
68.00	.9450	177.60	1.0144	175.56	.0254	36.80	.0241	33.45
65.00	.9545	177.01	1.0179	175.01	.0241	36.77	.0227	33.31
63.00	.9722	175.70	1.0255	173.72	.0224	34.50	.0210	30.84
61.00	.9899	174.21	1.0340	172.19	.0213	29.46	.0200	25.56
60.00	.9992	173.40	1.0408	171.34	.0209	26.31	.0197	22.29
56.00	1.0518	169.46	1.0986	167.14	.0223	15.27	.0211	11.02
55.00	1.0670	168.34	1.1040	165.93	.0232	12.93	.0220	8.68
51.00	1.1397	163.40	1.1900	160.51	.0286	3.74	.0272	-.52
50.00	1.1595	162.03	1.2010	158.98	.0305	1.30	.0289	-2.96
45.00	1.2705	154.05	1.3196	149.85	.0437	-12.78	.0414	-17.04
40.00	1.4056	143.43	1.4519	137.22	.0473	-31.71	.0638	-35.94
35.00	1.5603	128.73	1.6030	119.15	.1083	-58.42	.1026	-62.67
30.00	1.6497	108.28	1.6521	94.11	.1655	-95.42	.1569	-99.68
25.00	1.6925	82.27	1.4875	64.52	.2080	-139.53	.1972	-143.79
20.00	1.4934	55.23	1.1914	36.71	.2059	-180.11	.1952	-184.37
15.00	1.2587	31.98	.9472	13.76	.1816	-210.05	.1721	-214.30
10.00	1.0972	13.96	.8101	-4.32	.1614	-229.86	.1530	-234.12
5.00	1.0355	1.24	.7550	-17.30	.1532	-242.31	.1452	-246.56
0	1.0124	-6.15	.7345	-24.77	.1585	-249.34	.1503	-253.62

1R11 MAG/1R1 MAG = 1.99543E-01

1R11 ANGLE-1R1 ANGLE = -224.58662

Table E

APPENDIX A

Some Notes on Organization of the Waveguide Program.

a. Functional Dependence Chart

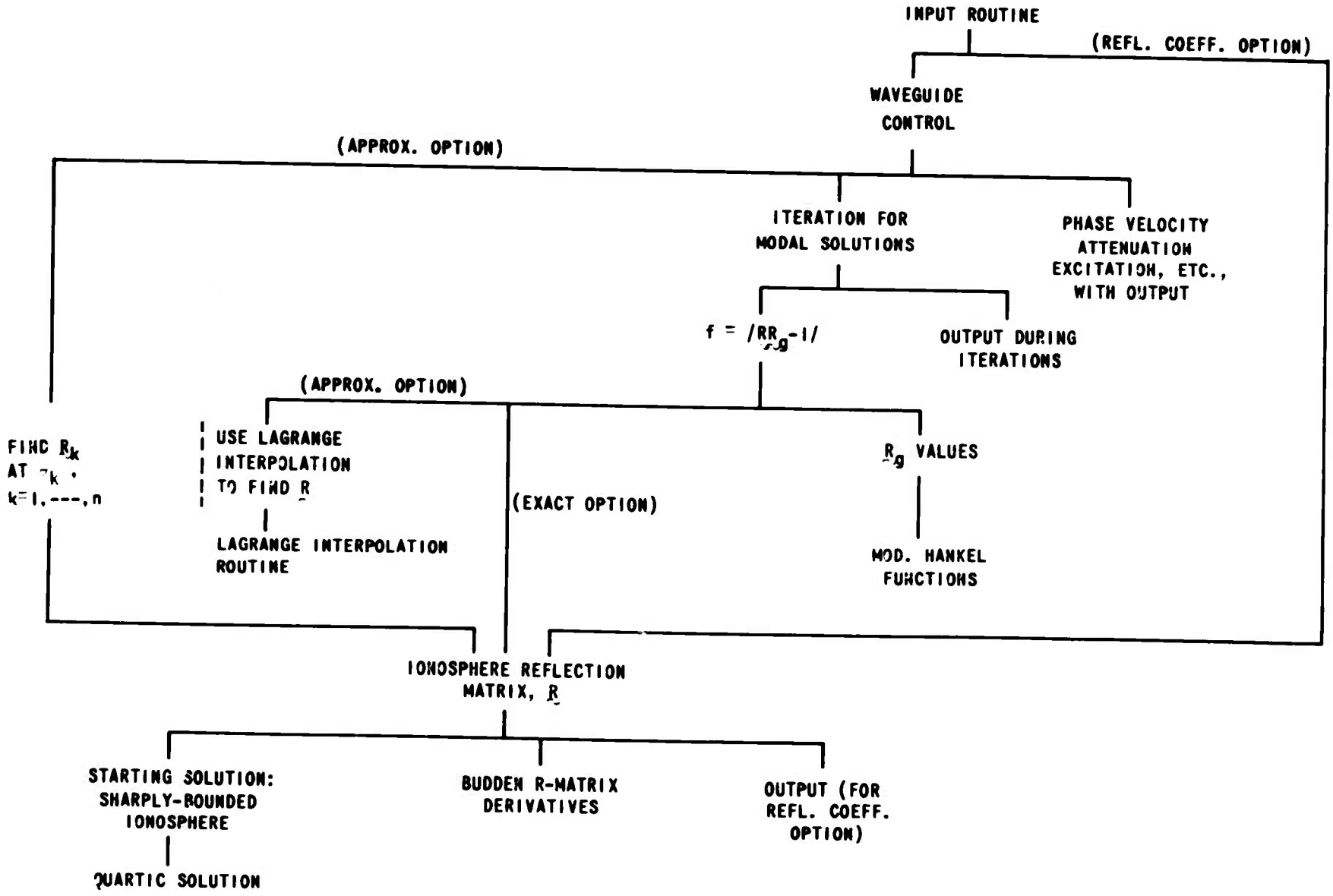
The general features of the waveguide program are illustrated on the next page in a kind of "functional dependence" chart (it is not a flow chart). Control in the program begins in PROGRAM INPUT which calls on the waveguide control routine, which in turn calls on the iteration routine, etc. Each routine is initiated by the one above it on the chart. The items in the chart correspond roughly to Fortran subprograms.

b. Program Input and Dummy

Program input is used at NELC for various program combinations other than the waveguide program described in this report. Hence some of the programming in this routine is not applicable to the waveguide program. Names of routines called on by INPUT, but not in this program deck, are programmed as entry points in DUMMY to make compilation possible.

The omitted programming consists of an alternate full-wave solution (a modified Inoue-Horowitz integration) and a scheme for fitting electron density profiles to reflection coefficient data. Description of these routines is beyond the scope of this report.

The program calls on ZERO CLK and PRINTIME. These are written in CDC 1064 assembly code and are used to time various segments of the program. It is probably best that the user substitute his own clock routine.



FUNCTIONAL DEPENDENCE CHART

```
 PROGRAM INPUT

 COMMON/INPUT/THETA,FREQ KHZ,AZIMUTH,DIP ANGLE,MAG FIELD,ALPHA,H,
$           D,REL PREC,TOP HT,LOWEST HT,IN EXTRA(88)
 COMMON/SP INPUT/NR SPEC,COEFF NU(3),EXP NU(3),CHARGE(3),M RATIO(3)
 COMMON/FLG INPUT/DEBUG,RC PRINT,POLY FLAG,INTEG UP,WF FLAG,
$           NEG ELECT,FLG EXTRA(19)
 COMMON/WG INPUT/TYPE ITER,MAX DELTA,EPSILON,EPSILON 0,SIGMA,
$           WGIN SKIP,
$           REFL HT,R POLY,T LIST(11),EIGEN(30),THETA INC,
$           T,PHI,DTHETA,WG EXTRA(5)
 COMMON/PF INPUT/NR V,NR F,
$           EXPER F(6),EXPER VAL(24),EXPER UNC(24),
$           DA,RMS TOL,NR TIMES,NR FRACT,FRACT(7),DA FACTOR,
$           PF EXTRA(32)
 COMMON/PR INPUT/NR A,HT0,HT UNIT,A(9),NR PTS,FN FACTOR,
$           PR EXTRA(11)
 COMMON/1/SKIP1,HT LIST(256),LOG N LIST(256,3)
 NAMELIST/DATA/THETA,FREQ KHZ,AZIMUTH,DIP ANGLE,MAG FIELD,ALPHA,H,
$           D,REL PREC,TOP HT,LOWEST HT,
$           COEFF NU,EXP NU,CHARGE,M RATIO,SPECIES,
$           DEBUG,POLY FLAG,NEG ELECT,
$           TYPE ITER,MAX DELTA,EPSILON,EPSILON 0,SIGMA,
$           REFL HT,R POLY,T LIST,EIGEN,THETA INC,T,PHI,
$           NR V,NR F,EXPER F,EXPER VAL,EXPER UNC,
$           DA,RMS TOL,NR TIMES,NR FRACT,FRACT,DA FACTOR,
$           NR A,HT0,HT UNIT,A,FN FACTOR

 COMPLEX THETA
 COMPLEX EIGEN,T LIST,DTHETA
 REAL MAG FIELD,LOWEST HT,M RATIO
 REAL MAX DELTA
 REAL LOG N LIST
 INTEGER DEBUG,RC PRINT,POLY FLAG,WF FLAG
 INTEGER ALPHANUM,SPECIES
 INTEGER TYPE ITER,R POLY
 DIMENSION ALPHANUM(10)

 DATA(ALPHA=0.0),(H=0.0)
 DATA(D=0.0)
 DATA(REL PREC=3.5)
 DATA(LOWEST HT=0.0)
 DATA(SPECIES=1),(NR SPEC=1)
 DATA(M RATIO(1)=1.0),(CHARGE(1)=-1.0)
 DATA(DEBUG=0)
 DATA(RC PRINT=1)
 DATA(POLY FLAG=0)
 DATA(INTEG UP=0)
 DATA(WF FLAG=0)
 DATA(NEG ELECT=0)

 DATA(TYPE ITER=0),(MAX DELTA=0.0)
 DATA(R POLY=0)
 DATA(T LIST=11((0.0,0.0)))
 DATA(EIGEN=30((0.0,0.0)))
 DATA(THETA INC=0.1)
```

```
      DATA(DTHETA=(0.05,0.01))

      DATA(NR V=9)
      DATA(NR F=3)
      DATA(DA=1.0)
      DATA(RMS TOL=1.05)
      DATA(NR TIMES=1)
      DATA(NR FRACT=7)
      DATA(FRACT=0.015625,0.03125,0.0625,0.125,0.25,0.5,1.0)
      DATA(DA FACTOR=1.0)

      DATA(NR A=6)
      DATA(HT0=60.0),(HT UNIT=15.0)
      DATA(A=9(0.0))
      DATA(FN FACTOR=1.0E03)


      PRINT 100
  100 FORMAT(1H1)

   10 NW = INLIST(DATA,N)
      IF(NW .EQ. 7HPROFILE) GO TO 20
      IF(NW .EQ. 7HREVERSE) GO TO 30
      IF(NW .EQ. 7HNEGIONS) GO TO 35
      IF(NW .EQ. 2HRC) GO TO 40
      IF(NW .EQ. 8HINITIALA) GO TO 45
      IF(NW .EQ. 2HWF) GO TO 50
      IF(NW .EQ. 8HWAVEGUID) GO TO 60
      IF(NW .EQ. 6HENCOEF) GO TO 70
      IF(NW .EQ. 7HFITPROF) GO TO 71
      IF(NW .EQ. 8HENUNCERT) GO TO 72
      IF(NW .EQ. 4HQUIT) GO TO 99
      GO TO 90

   20 READ 201,(ALPHANUM(J),J=1,10)
      PRINT 202,(ALPHANUM(J),J=1,10)
      K = SPECIES
      IF(K .GT. NR SPEC) NR SPEC = K
      L = 1
   21 READ 201,(ALPHANUM(J),J=1,10)
  201 FORMAT(10A8)
      PRINT 202,(ALPHANUM(J),J=1,10)
  202 FORMAT(1H ,10A8)
      IF(ALPHANUM(1) .EQ. 8H99999999) GO TO 22
      DECODE(21,203,ALPHANUM) HT,EN
  203 FORMAT(F7.2,5X,E9.2)
      IF(K .NE. 1 .AND. HT .NE. HT LIST(L)) GO TO 90
      HT LIST(L) = HT
      IF(L .NE. 1 .AND. HT LIST(L) .GE. HT LIST(L-1)) GO TO 90
      IF(EN .EQ. 0.0) EN = 1.0E-40
      LOG N LIST(L,K) = LOGF(EN)
      L = L+1
      GO TO 21
   22 TOP HT = HT LIST(1)
      LOWEST HT = HT LIST(L-1)
      NR PTS = L-1
      INTEG UP = 0
      GO TO 10
```

```
   30 L HALF = NR PTS/2
      DO 31 L=1,L HALF
      M = NR PTS+1-L
      TEMP = HT LIST(L) $ HT LIST(L) = HT LIST(M) $ HT LIST(M) = TEMP
      TEMP = LOG N LIST(L,1) $ LOG N LIST(L,1) = LOG N LIST(M,1)
                LOG N LIST(M,1) = TEMP
   31 CONTINUE
      DO 32 L=1,NR PTS
   32 HT LIST(L) = -HT LIST(L)
      INTEG UP = 1-INTEG UP
      TEMP = TOP HT $ TOP HT = -LOWEST HT $ LOWEST HT = -TEMP
      GO TO 10

   35 DO 36 L=1,NR PTS
      EN = EXPF(LOG N LIST(L,2))-EXPF(LOG N LIST(L,1))
      IF(EN .EQ. 0.0) EN = 1.0E-40
      LOG N LIST(L,3) = LOGF(EN)
   36 PRINT 301,HT LIST(L),EN
  301 FORMAT(1H ,F7.2,5X,E9.2)
      NR SPEC = 3
      GO TO 10

C RC
   40 CALL ZERO CLK
      D = LOWEST HT
      CALL INTEG
      CALL PRINTIME
      PRINT 100
      GO TO 10

   45 D = LOWEST HT
      CALL INITIAL A
      GO TO 10

C WAVEFIELDS
   50 CALL ZERO CLK
      WF FLAG = 1
      D = LOWEST HT
      CALL INTEG
      CALL BACKWARD
      WF FLAG = 0
      CALL PRINTIME
      GO TO 10

C WAVEGUID
   60 CALL WAVEGUID
      GO TO 10

C PROFILE FITTING
   70 CALL EN COEF
      GO TO 10

   71 D = 0.0
      CALL FIT PROF
      GO TO 10

   72 CALL EN UNCERT
      GO TO 10
```

```
C ERROR EXIT
   90 PRINT 900
  900 FORMAT(1H0,18HERROR IN DATA DECK)

   99 CONTINUE

      END
```

```
      SUBROUTINE DUMMY
C FOR BUDDEN WAVEGUIDE

      ENTRY FIT PROF
      ENTRY EN UNCERT
      ENTRY EN COEF
      ENTRY BACKWARD
      ENTRY INITIAL A
      RETURN

      END
```

```
   FUNCTION CPLX FCT(ARG)

   COMPLEX CPLX FCT,ARG,I
   REAL NEG
   DATA(SAVE REAL=1.0E99),(SAVE IMAG=1.0E99)
   DATA(I=(0.0,1.0))
   DATA(PI=3.1415926536)


   ENTRY CPLX SQRT
   ARG REAL = ARG $ ARG IMAG = -I*ARG
   RHO = SQRTF(ARG REAL*ARG REAL+ARG IMAG*ARG IMAG)
   ABS REAL = ABSF(ARG REAL)
   IF(RHO .LT. ABS REAL) RHO = ABS REAL
   IF(ARG IMAG .GE. 0.0) 1,2
 1 CPLX FCT = SQRTF((RHO+ARG REAL)*0.5)+I*SQRTF((RHO-ARG REAL)*0.5)
   RETURN
 2 CPLX FCT = -SQRTF((RHO+ARG REAL)*0.5)+I*SQRTF((RHO-ARG REAL)*0.5)
   RETURN

   ENTRY CPLX LOGF
   ARG REAL = ARG $ ARG IMAG = -I*ARG
   RHO = SQRTF(ARG REAL*ARG REAL+ARG IMAG*ARG IMAG)
   IF(RHO .EQ. 0.0) GO TO 13
   ABS REAL = ABSF(ARG REAL)
   IF(RHO .LT. ABS REAL) RHO = ABS REAL
   IF(ARG IMAG .GE. 0.0) 11,12
11 CPLX FCT = LOGF(RHO)+I*ACOSF(ARG REAL/RHO) $ RETURN
12 CPLX FCT = LOGF(RHO)+I*(2.0*PI-ACOSF(ARG REAL/RHO))
   RETURN
13 CPLX FCT = -200.0 $ RETURN

   ENTRY CPLX EXPF
   ARG REAL = ARG $ ARG IMAG = -I*ARG
   IF(ARG IMAG .EQ. 0.0)21,22
21 CPLX FCT = EXPF(ARG REAL) $ RETURN
22 IF(ARG REAL .EQ. 0.0)23,24
23 CPLX FCT = COSF(ARG IMAG)+I*SINF(ARG IMAG) $ RETURN
24 CPLX FCT = EXPF(ARG REAL)*(COSF(ARG IMAG)+I*SINF(ARG IMAG))
   RETURN

   ENTRY CPLX COSF
   ARG REAL = ARG $ ARG IMAG = -I*ARG
   IF(ARG IMAG .EQ. 0.0)31,32
31 CPLX FCT = COSF(ARG REAL) $ RETURN
32 IF(ARG REAL .EQ. SAVE REAL .AND. ARG IMAG .EQ. SAVE IMAG) GO TO 33
   SAVE REAL = ARG REAL $ SAVE IMAG = ARG IMAG
   COS = COSF(ARG REAL)
   SIN = SINF(ARG REAL)
   POS = EXPF(ARG IMAG) $ NEG = 1.0/POS
   COSH = (POS+NEG)*0.5
   SINH = (POS-NEG)*0.5
33 CPLX FCT = COS*COSH-I*SIN*SINH
   RETURN

   ENTRY CPLX SINF
   ARG REAL = ARG $ ARG IMAG = -I*ARG
   IF(ARG IMAG .EQ. 0.0)41,42
```

```
41 CPLX FCT = SINF(ARG REAL) $ RETURN
42 IF(ARG REAL .EQ. SAVE REAL .AND. ARG IMAG .EQ. SAVE IMAG) GO TO 43
   SAVE REAL = ARG REAL $ SAVE IMAG = ARG IMAG
   COS = COSF(ARG REAL)
   SIN = SINF(ARG REAL)
   POS = EXPF(ARG IMAG) $ NEG = 1.0/POS
   COSH = (POS+NEG)*0.5
   SINH = (POS-NEG)*0.5
43 CPLX FCT = SIN*COSH+I*COS*SINH
   RETURN

   ENTRY CONJ
   ARG REAL = ARG $ ARG IMAG = -I*ARG
   CPLX FCT = ARG REAL-I*ARG IMAG
   RETURN

   END
```

```
SUBROUTINE MAG ANGLE(ARG,MAG,ANGLE)

COMPLEX ARG,I
REAL MAG
DATA(I=(0.0,1.0))
DATA(RAD TO DEG=57.29577951)


ARG REAL = ARG $ ARG IMAG = -I*ARG
MAG = SQRTF(ARG REAL*ARG REAL+ARG IMAG*ARG IMAG)
COS = ARG REAL/MAG
IF(COS .GT. 1.0 .AND. COS .LT. 1.01) COS = 1.0
IF(COS .LT. -1.0 .AND. COS .GT. -1.01) COS = -1.0
ANGLE = ACOSF(COS)*RAD TO DEG
IF(ARG IMAG .LT. 0.0) ANGLE = 360.0-ANGLE
RETURN

END
```

```
      SUBROUTINE QUARTIC(B3,B2,B1,B0,Q,DEBUG)

      COMPLEX B3,B2,B1,B0,Q,
     $          B3 SQ,H,I,G,H PRIME,G PRIME,
     $          SQ ROOT,P PLUS,P,LOG P,
     $          CUBE RT0,CUBE RT1,CUBE RT2,OMEGA1,OMEGA2,
     $          ROOT P,ROOT Q,ROOT R,
     $          FUNCTION,CPLX I,
     $          CPLX SQRT,CPLX LOGF,CPLX EXPF
      INTEGER DEBUG,ERR FLAG
      REAL MAG PLUS,MAG MINUS,MAG F
      DIMENSION Q(4),P RI(2),FUNCTION(4)
      DATA(OMEGA1=(-0.5,0.8660254038)),(OMEGA2=(-0.5,-0.8660254038))
      DATA(CPLX I=(0.0,1.0))
      DATA(TOL=1.0E-02)
      EQUIVALENCE(P,P RI)


      B3 SQ = B3**2
      H = B2-B3 SQ
      I = B0-4.0*B3*B1+3.0*B2**2
      G = B1+B3*(-3.0*B2+2.0*B3 SQ)
      H PRIME = -I/12.0
      G PRIME = -G**2/4.0-H*(H**2+3.0*H PRIME)

      SQ ROOT = CPLX SQRT(G PRIME**2+4.0*H PRIME**3)
      P = (-G PRIME+SQ ROOT)*0.5
      MAG PLUS = ABSF(P RI(1))+ABSF(P RI(2))
      P PLUS = P
      P = (-G PRIME-SQ ROOT)*0.5
      MAG MINUS = ABSF(P RI(1))+ABSF(P RI(2))
      IF(MAG PLUS .GT. MAG MINUS) P = P PLUS
      LOG P = CPLX LOGF(P)
      CUBE RT0 = CPLX EXPF(LOG P/3.0)
      CUBE RT1 = OMEGA1*CUBE RT0
      CUBE RT2 = OMEGA2*CUBE RT0

      ROOT P = CPLX SQRT(CUBE RT0-H PRIME/CUBE RT0-H)
      ROOT Q = CPLX SQRT(CUBE RT1-H PRIME/CUBE RT1-H)
      ROOT R = CPLX SQRT(CUBE RT2-H PRIME/CUBE RT2-H)
      SIGN = -ROOT P*ROOT Q*ROOT R*2.0/G
      IF(SIGN .LT. 0.0) ROOT R = -ROOT R
      Q(1) = +ROOT P+ROOT Q+ROOT R-B3
      Q(2) = +ROOT P-ROOT Q-ROOT R-B3
      Q(3) = -ROOT P+ROOT Q-ROOT R-B3
      Q(4) = -ROOT P-ROOT Q+ROOT R-B3

      DO 32 N=1,4
      QR = Q(N) $ ABS QR = ABSF(QR)
      QI = -CPLX I*Q(N) $ ABS QI = ABSF(QI)
      IF(ABS QR*TOL .LT. ABS QI) GO TO 31
      QI = CPLX I*((((QR+4.0*B3)*QR+6.0*B2)*QR+4.0*B1)*QR+B0)
     $          /(((4.0*QR+12.0*B3)*QR+12.0*B2)*QR+4.0*B1)
   31 IF(ABS QI*TOL .LT. ABS QR) GO TO 32
      QR = ((((QI-CPLX I*4.0*B3)*QI-6.0*B2)*QI+CPLX I*4.0*B1)*QI+B0)
     $          /(((CPLX I*4.0*QI+12.0*B3)*QI-CPLX I*12.0*B2)*QI-4.0*B1)
   32 Q(N) = QR+CPLX I*QI
```

```
      IF(DEBUG .EQ. 0) RETURN

      ERR FLAG = 0
      DO 51 N=1,4
      IF(DEBUG .GE. 2) PRINT 501,N,Q(N)
  501 FORMAT(1H+,5HROOT ,I1,3H = ,C(E12.5,E13.5))
      FUNCTION(N) = ((((Q(N)+4.0*B3)*Q(N)+6.0*B2)*Q(N)+4.0*B1)*Q(N)+B0)
     $           /B0
      IF(DEBUG .GE. 2) PRINT 502,N,FUNCTION(N)
  502 FORMAT(1H ,40X,9HFUNCTION ,I1,C(E11.3,E11.3))
      CALL MAG ANGLE(FUNCTION(N),MAG F,ANGLE F)
      IF(MAG F .LT. 0.10) GO TO 51
      PRINT 503,N,MAG F
  503 FORMAT(1H ,5HROOT ,I1,23H IS NO GOOD    MAG F =  ,E10.3)
      ERR FLAG = 1
   51 CONTINUE
      IF(ERR FLAG .EQ. 1) STOP
      RETURN

      END
```

```
      SUBROUTINE WAVEGUID

      COMMON/INPUT/THETA,IN OMIT(98)
      COMMON/FLG INPUT/FLG SKIP,RC PRINT,FLG OMIT(23)
      COMMON/WG INPUT/WGIN SKIP(7),R POLY,WGIN OMIT(22),EIGEN(30),
     $           WGIN NO(10)
      COMPLEX THETA,EIGEN
      INTEGER RC PRINT,R POLY


      RC PRINT = 0
      IF(R POLY .EQ. 1) CALL GEN R POLY

      INDEX = 1
   11 THETA = EIGEN(INDEX)
      CALL ITERATE
      CALL COMP PROC
      EIGEN RL = EIGEN(INDEX+1)
      IF(EIGEN RL .EQ. 0.0) GO TO 21
      INDEX = INDEX+1
      GO TO 11

   21 RC PRINT = 1
      RETURN

      END
```

```
      SUBROUTINE ITERATE

      COMMON/INPUT/THETA,IN OMIT(98)
      COMMON/WG INPUT/TYPE ITER,MAX DELTA,WGIN SKIP(88),THETA INC,
     $           WGIN OMIT(2),DTHETA,WGIN NO(5)
      COMMON/R/R SKIP(18),R11,R22,R OMIT(4)
      COMMON/F/F,DFDTHETA,R BAR11,R BAR22
      COMPLEX THETA,F0,F,DFDTHETA,DELTHETA,I,R11,R22,R BAR11,R BAR22,
     $           EIGEN,DTHETA
      REAL MAX DELTA,LUB REAL,LUB IMAG
      INTEGER TYPE ITER,RC PRINT,R POLY,FIRST,FIXED INC
      DATA(MAX ITER=20),(LUB REAL=0.05),(LUB IMAG=0.005)
      DATA(I=(0.0,1.0))


      PRINT 101,TYPE ITER
  101 FORMAT(1H1,14HITERATION TYPE ,I3)
      PRINT 102,MAX DELTA
  102 FORMAT(1H ,12HMAX DELTA =  ,F5.2)
      FIRST = 1
      NR ITER = 0

   11 THETA = THETA-DTHETA
      CALL COMPUTE F
      F0 = F
      THETA = THETA+DTHETA
      CALL COMPUTE F
      DFDTHETA = (F-F0)/DTHETA
      DELTHETA = -F/DFDTHETA
      FIXED INC = 1
      GO TO 20

   15 F0 = F
      CALL COMPUTE F
      DFDTHETA = (F-F0)/DELTHETA
      DELTHETA = -F/DFDTHETA
      FIXED INC = 0

   20 IF(FIRST .EQ. 0) GO TO 21
      CALL PRINT RS $ CALL PRT THETA $ FIRST = 0
   21 DEL REAL = DELTHETA $ ABS REAL = ABSF(DEL REAL)
      IF(ABS REAL .GT. THETA INC) DELTHETA = DELTHETA*THETA INC/ABS REAL
      DEL IMAG = -I*DELTHETA $ ABS IMAG = ABSF(DEL IMAG)
      IF(ABS IMAG .GT. THETA INC) DELTHETA = DELTHETA*THETA INC/ABS IMAG
      THETA = THETA+DELTHETA
      CALL PRT THETA
      NR ITER = NR ITER+1
      IF(NR ITER .GT. MAX ITER) GO TO 30
      DEL REAL = DELTHETA $ DEL IMAG = -I*DELTHETA
      ABS REAL = ABSF(DEL REAL) $ ABS IMAG = ABSF(DEL IMAG)
      IF(ABS REAL .LE. LUB REAL .AND. ABS IMAG .LE. LUB IMAG) GO TO 30
      IF(ABS REAL .LE. MAX DELTA .AND. ABS IMAG .LE. MAX DELTA) 15,11

   30 IF(FIXED INC .EQ. 1) GO TO 31
      THETA = THETA-DTHETA
      CALL COMPUTE F
      F0 = F
      THETA = THETA+DTHETA
```

```
      CALL COMPUTE F
      DFDTHETA = (F-F0)/DTHETA
   31 IF(TYPE ITER .EQ. 1) DFDTHETA = (R BAR22*R22-1.0)*DFDTHETA
      IF(TYPE ITER .EQ. 2) DFDTHETA = (R BAR11*R11-1.0)*DFDTHETA

      CALL PRINT RS
      PRINT 501,NR ITER
  501 FORMAT(1H0,23HITERATIONS PERFORMED =  ,I2)
      PRINT 502,DELTHETA
  502 FORMAT(1H0,11HDELTHETA =  ,C(F11.7,F11.7))
      PRINT 503,F
  503 FORMAT(1H0,13HF FUNCTION =  ,C(F13.5,F13.5))
      RETURN

      END
```

```
      SUBROUTINE COMPUTE F

      COMMON/WG INPUT/TYPE ITER,WGIN SKIP(6),R POLY,WGIN OMIT(92)
      COMMON/R/R SKIP(18),R11,R22,R12,R21
      COMMON/F/F,F SKIP(2),R BAR11,R BAR22
      COMPLEX R11,R22,R12,R21,F,R BAR11,R BAR22
      INTEGER TYPE ITER,R POLY


      CALL R BARS

      IF(R POLY .EQ. 0) CALL INTEG
      IF(R POLY .EQ. 1) CALL USE POLY

      IF(TYPE ITER .EQ. 0) F = (R BAR11*R11-1.0)*(R BAR22*R22-1.0)
     $           -R BAR11*R BAR22*R12*R21
      IF(TYPE ITER .EQ. 1) F = R BAR11*R11-1.0
      IF(TYPE ITER .EQ. 2) F = R BAR22*R22-1.0
      RETURN

      END
```

```
      SUBROUTINE R BARS

      COMMON/INPUT/THETA,FREQ KHZ,IN SKIP(3),ALPHA,H,D,IN OMIT(91)
      COMMON/WG INPUT/WG SKIP(2),EPSILON,EPSILON 0,SIGMA,WG OMIT(95)
      COMMON/F/F SKIP(4),R BAR11,R BAR22
      COMMON/RB CP/S H FACTOR
      COMPLEX THETA,S H FACTOR,I,NG SQ,C,S,SQ ROOT,RATIO RT,
     $          PD,QD,H1 D,H2 D,H1 PRIME D,H2 PRIME D,CAP H1 D,CAP H2 D,
     $          P0,Q0,H1 0,H2 0,H1 PRIME 0,H2 PRIME 0,CAP H1 0,CAP H2 0,
     $          A1ST,A2ND,A1,A2,A3,A4,R BAR11,R BAR22,
     $          C POWER,Z1,Z2,C TEMP,
     $          CPLX SQRT,CPLX EXPF,CPLX SINF,CPLX COSF
      REAL K,LOG K OVR A,K OVER A OT,K OVER A TT,ND SQ,N0 SQ
      DATA(PI=3.141592653)
      DATA(VEL LIGHT=2.997928E05)
      DATA(I=(0.0,1.0))
      DATA(DEG TO RAD=0.01745329252)
      DATA(TEST TH IM=10.0)


      OMEGA = 2.0*PI*FREQ KHZ*1000.0
      K = OMEGA/VEL LIGHT
      NG SQ = (EPSILON-I*SIGMA/OMEGA)/EPSILON 0

      C = CPLX COSF(THETA*DEG TO RAD)
      S = CPLX SINF(THETA*DEG TO RAD)

      LOG K OVR A = LOGF(K/ALPHA)
      K OVER A OT = EXPF(LOG K OVR A/3.0)
      K OVER A TT = K OVER A OT**2
      A OVER K OT = 1.0/K OVER A OT
      A OVER K TT = A OVER K OT**2

      ND SQ = 1.0-ALPHA*(H-D)
      N0 SQ = 1.0-ALPHA*H
      SQ ROOT = -CPLX SQRT(NG SQ-S**2)
      RATIO RT = N0 SQ/NG SQ*SQ ROOT

      PD = K OVER A TT*(1.0-ALPHA*(H-D)-S**2)
      P0 = K OVER A TT*(1.0-ALPHA*H-S**2)
      Q TERM = K OVER A TT*(ALPHA/K)**2/4.0
      QD = PD-Q TERM
      Q0 = P0-Q TERM

      THETA IM = -I*THETA
      IF(-THETA IM .GT. TEST TH IM .OR. D .EQ. 0.0) GO TO 50
      CALL MD HANKEL(QD,H1 D,H2 D,H1 PRIME D,H2 PRIME D)
      CALL MD HANKEL(Q0,H1 0,H2 0,H1 PRIME 0,H2 PRIME 0)

      CAP H1 D = H1 PRIME D+A OVER K TT*H1 D/2.0
      CAP H1 0 = H1 PRIME 0+A OVER K TT*H1 0/2.0
      CAP H2 D = H2 PRIME D+A OVER K TT*H2 D/2.0
      CAP H2 0 = H2 PRIME 0+A OVER K TT*H2 0/2.0

      A1ST = CAP H2 0-I*RATIO RT*K OVER A OT*H2 0
      A2ND = CAP H1 0-I*RATIO RT*K OVER A OT*H1 0
      A1 = C*ND SQ*(-H1 D*A1ST+H2 D*A2ND)
      S H FACTOR = (-A1ST*H1 0+A2ND*H2 0)/(-A1ST*H1 D+A2ND*H2 D)
```

```
      A1ST = -I*A OVER K OT*CAP H2 0-RATIO RT*H2 0
      A2ND = -I*A OVER K OT*CAP H1 0-RATIO RT*H1 0
      A2 = -CAP H1 D*A1ST+CAP H2 D*A2ND
      R BAR11 = (A1-A2)/(A1+A2)

      CALL MD HANKEL(PD,H1 D,H2 D,H1 PRIME D,H2 PRIME D)
      CALL MD HANKEL(P0,H1 0,H2 0,H1 PRIME 0,H2 PRIME 0)

      A1ST = I*A OVER K OT*H2 PRIME 0+SQ ROOT*H2 0
      A2ND = I*A OVER K OT*H1 PRIME 0+SQ ROOT*H1 0
      A3 = -H1 PRIME D*A1ST+H2 PRIME D*A2ND
      A1ST = H2 PRIME 0-I*K OVER A OT*SQ ROOT*H2 0
      A2ND = H1 PRIME 0-I*K OVER A OT*SQ ROOT*H1 0
      A4 = C*(-H1 D*A1ST+H2 D*A2ND)
      R BAR22 = (A3+A4)/(A4-A3)
      RETURN

C FLAT EARTH
   50 C POWER = CPLX EXPF(-2.0*I*K*D*C)
      R BAR11 = (NG SQ*C-SQ ROOT)/(NG SQ*C+SQ ROOT)*C POWER
      R BAR22 = (C-SQ ROOT)/(C+SQ ROOT)*C POWER
      Z1 = C+SQ ROOT/NG SQ
      Z2 = -C+SQ ROOT/NG SQ
      C POWER = CPLX EXPF(I*K*D*C)
      C TEMP = (Z1-Z2)/(Z1*C POWER-Z2/C POWER)
      S H FACTOR = C TEMP**2
      RETURN

      END
```

```
 SUBROUTINE COMP PROC

 COMMON/INPUT/THETA,FREQ KHZ,IN SKIP(3),ALPHA,H,D,IN OMIT(91)
 COMMON/WG INPUT/WGIN SKIP(3),EPSILON 0,WGIN OMIT(2),
$           REFL HT,WGIN NO(93)
 COMMON/F/F SKIP(2),DFDTHETA,R BAR11,R BAR22
 COMMON/R/R SKIP(18),R11,R22,R12,R21
 COMMON/RB CP/S H FACTOR
 COMPLEX I,S,SQRT S,E I PI OV 4,STORE 1,STORE 2,
$           S H FACTOR,SCRIPT H,
$           DFDTHETA,THETA,CPLX SQRT,CPLX SINF,S THETA P,C THETA P,
$           R11,R22,R12,R21,R BAR11,R BAR22,
$           EXTRA,WAITS EF,DENOM,POLAR
 REAL K,LOGE10,IMAG
 DATA(I=(0.0,1.0))
 DATA(DEG TO RAD=0.01745329252),(RAD TO DEG=57.29577951)
 DATA(VEL LIGHT=2.997928E05)
 DATA(PI=3.141592653)
 DATA(E I PI OV 4=(0.7071067812,0.7071067812))
 DATA(LOGE10=2.302585093)


 K = 2.0*PI*FREQ KHZ*1000.0/VEL LIGHT
 CAP K = 1.0/(1.0-ALPHA*H/2.0)
 S = CPLX SINF(THETA*DEG TO RAD)
 S THETA P = CAP K*S
 STP REAL = S THETA P
 STP IMAG = -I*S THETA P
 PHASE VEL = VEL LIGHT/STP REAL
 V OVER C = PHASE VEL/VEL LIGHT
 ATTEN = -8.6858896*K*STP IMAG*1000.0

 SCRIPT H = EXPF(-ALPHA*D/2.0)*S H FACTOR
 SQRT S = -CPLX SQRT(S)
 STORE 1 = K*K*SQRTF(K)*S*SQRT S*E I PI OV 4
$           /(2.0*EPSILON 0*SQRTF(2.0*PI))
 STORE 2 = (1.0+R BAR11)**2*(1.0-R BAR22*R22)*SCRIPT H**2
$           /(DFDTHETA/DEG TO RAD*R BAR11)
 EXTRA = STORE 2*S
 WAITS EF = -I*K*REFL HT/2.0*EXTRA

 CALL MAG ANGLE(EXTRA,EXTRA MAG,EXTRA ANG)
 EXTRA ANG = EXTRA ANG*DEG TO RAD
 CALL MAG ANGLE(WAITS EF,WAITS MAG,WAITS ANG)
 WAITS ANG = WAITS ANG*DEG TO RAD
 WAIT EF DB = 20.0*LOGF(WAITS MAG)/LOGE10

 DENOM = R BAR11*R11-1.0
 REAL  = DENOM
 IMAG = -I*DENOM
 IF(REAL .EQ. 0.0 .AND. IMAG .EQ. 0.0) DENOM = 1.0E-150
 POLAR = -R BAR11*R12/DENOM
 CALL MAG ANGLE(POLAR,POLAR MAG,POLAR ANG)

 C THETA P = CPLX SQRT(1.0-S THETA P**2)
 CTP REAL = C THETA P
 COS = SQRTF(CTP REAL**2-STP IMAG**2)
 THETA P RL = ACOSF(COS)*RAD TO DEG
```

```
      THETA P IM = LOGF((CTP REAL+STP IMAG)/COS)*RAD TO DEG


      PRINT 100
  100 FORMAT(1H0,////)
      PRINT 101,PHASE VEL
  101 FORMAT(1H ,2X,17HPHASE VELOCITY = ,E12.5,11H KM PER SEC)
      PRINT 102,V OVER C
  102 FORMAT(1H ,2X,24HPHASE VELOCITY OVER C = ,F9.5)
      PRINT 103,ATTEN
  103 FORMAT(1H0,2X,14HATTENUATION = ,E12.5,3H DB)
      PRINT 104,EXTRA MAG
  104 FORMAT(1H0,2X,12HEXTRA MAG = ,E12.5)
      PRINT 105,EXTRA ANG
  105 FORMAT(1H ,2X,14HEXTRA ANGLE = ,F10.5,4H RAD)
      PRINT 112,WAITS ANG
  112 FORMAT(1H0,2X,35HPHASE OF WAITS EXCITATION FACTOR = ,F10.5,4H RAD)
      PRINT 113,WAIT EF DB
  113 FORMAT(1H ,2X,32HWAITS EXCITATION FACTOR IN DB = ,E12.5,3H DB)
      PRINT 116,POLAR MAG
  116 FORMAT(1H0,2X,19HPOLARIZATION MAG = ,E12.5)
      PRINT 117,POLAR ANG
  117 FORMAT(1H ,2X,21HPOLARIZATION ANGLE = ,F10.5,4H DEG)
      PRINT 118,THETA P RL,THETA P IM
  118 FORMAT(1H0,2X,14HTHETA PRIME = ,2F8.3,4H DEG)
      PRINT 119
  119 FORMAT(1H1)
      RETURN

      END
```

```
      SUBROUTINE WG OUTPUT

      COMMON/INPUT/THETA,THETA IMG,IN OMIT(98)
      COMMON/R/R SKIP(18),R(4)
      COMMON/F/F,DFDTHETA,R BAR(2)
      COMPLEX R BAR,R,ALL RS,I,F,DFDTHETA
      REAL IMAG PART,MAG
      DATA(I=(0.0,1.0))
      DIMENSION ALL RS(8),MAG(8),ANGLE(8),REAL PART(8),IMAG PART(8)


      ENTRY PRINT RS
      DO 11 J=1,2
   11 ALL RS(J) = R BAR(J)
      DO 12 J=1,4
   12 ALL RS(J+2) = R(J)
      ALL RS(7) = R BAR(1)*R(1)
      ALL RS(8) = R BAR(2)*R(2)

      DO 13 J=1,8
      CALL MAG ANGLE(ALL RS(J),MAG(J),ANGLE(J))
      REAL PART(J) = ALL RS(J)
   13 IMAG PART(J) = -I*ALL RS(J)

      PRINT 200
  200 FORMAT(1H0,8X,5HTHETA,5X,9H11R11 BAR,6X,7H1R1 BAR,8X,5H11R11,
     $           10X,3H1R1,9X,4H1R11,9X,4H11R1,2X,11HR11 BAR*R11,
     $           4X,9HR1 BAR*R1/)
      PRINT 201,THETA,(REAL PART(J),J=1,8)
  201 FORMAT(1H ,4HREAL,F9.3,1X,8E13.5)
      PRINT 202,THETA IMG,(IMAG PART(J),J=1,8)
  202 FORMAT(1H ,4HIMAG,F9.3,1X,8E13.5)
      PRINT 203,(MAG(J),J=1,8)
  203 FORMAT(1H ,3HMAG,11X,8F13.5)
      PRINT 204,(ANGLE(J),J=1,8)
  204 FORMAT(1H ,5HANGLE, 9X,8F13.5)
      RETURN

      ENTRY PRT THETA
      CALL MAG ANGLE(F,F MAG,F ANGLE)
      CALL MAG ANGLE(DFDTHETA,DFDT MAG,DFDT ANGL)
      PRINT 500,THETA,THETA IMG,F MAG,DFDT MAG
  500 FORMAT(1H0,8HTHETA = ,2F10.3,10X,8HF MAG = ,E10.3,5X,
     $           15HDFDTHETA MAG = ,E10.3)
      RETURN

      END
```

```
      SUBROUTINE MD HANKEL(Z,H1,H2,H1 PRIME,H2 PRIME)

      COMPLEX Z,I,H1,H2,H1 PRIME,H2 PRIME,ZCUBED,Z POWER,TERM1,TERM2,
     $     TERM3,TERM4,TERM5,TERM6,SUM1,SUM2,SUM3,SUM4,SUM5,SUM6,
     $     SUM7,SUM8,F,G,F PRIME,G PRIME,ONE RT Z,CPLX SQRT,
     $     Z FOURTH,SQRT Z CUB,   EXP1,EXP2,EXP3,EXP4,EXP5,
     $     CPLX EXPF,BETA,RT Z,GM2F,GPMFP,I POWER,M I POWER,
     $     Z M3HALVS,Z M3HAFS M

      DIMENSION A(23),B(23),C(23),D(23),CAP(14)

      DATA(A=  0.9304 3671  693,  31.0145 5723  097, 206.7637 1487  316,
     $      574.3436 5242  545, 870.2176 5519  008, 828.7787 1922  864,
     $      541.6854 3740  434, 257.9454 4638  302,  93.4584 9506  631,
     $       26.6263 5187  074,   6.1210 0043 0056,   1.1592 8038 4480,
     $        0.1840 1275 9441,   0.0248 3303 0964,   0.0028 8420 8010,
     $        0.0002 9133 4142,   0.0000 2582 7495,   0.0000 0202 5686,
     $        0.0000 0014 1557,   0.0000 0000 8870,   0.0000 0000 0501,
     $        0.0000 0000 0026,   0.0000 0000 0001)

      DATA(B=  0.6782 9872  514,  11.3049 7875  240,  53.8332 3215  431,
     $      119.6294 0478  735, 153.3710 3177  865, 127.8091 9314  888,
     $       74.7422 1821  572,  32.3559 3862  152,  10.7853 1287  384,
     $        2.8532 5737  403,   0.6136 0373 6351,   0.1093 7678   98,
     $        0.0164 2293 9955,   0.0021 0550 5122,   0.0002 3316 7788,
     $        0.0000 2252 8289,   0.0000 0191 5671,   0.0000 0014 4470,
     $        0.0000 0000 9729,   0.0000 0000 0589,   0.0000 0000 0032,
     $        0.0000 0000 0002,   0.0000 0000 0000)

      DATA(C=  0.4652 1835  846,   6.2029 1144  619,  25.8454 6435  915,
     $       52.2130 5931  140,  62.1584 0394  215,  48.7516 8936  639,
     $       27.0842 7187  022,  11.2150 1940  796,   3.5945 5750  255,
     $        0.9181 5006  451,   0.1912 8126 3439,   0.0331 2229 6699,
     $        0.0048 4244 1038,   0.0006 0568 3682,   0.0000 6555 0182,
     $        0.0000 0619 8599,   0.0000 0051 6550,   0.0000 0003 8220,
     $        0.0000 0000 2528,   0.0000 0000 0150,   0.0000 0000 0008,
     $        0.0000 0000 0000,   0.0000 0000 0000)

      DATA(D=  0.6782 9872  514,  45.2199 1500  962, 376.8326 2508  015,
     $     1196.2940 4787  350,1993.8234 1312  250,2044.9470 9038  206,
     $     1420.1021 4609  865, 711.8306 4967  351, 269.6328 2184  603,
     $       79.8912 0647  290,  19.0217 1582 6880,   3.7188 1052 3339,
     $        0.6076 4877 8323,   0.0842 2020 4896,   0.0100 2621 4869,
     $        0.0010 3630 1278,   0.0000 9386 7869,   0.0000 0751 2435,
     $        0.0000 0053 5074,   0.0000 0003 4135,   0.0000 0000 1962,
     $        0.0000 0000 0102,   0.0000 0000 0005)

      DATA(CAP=0.1041 6666 6666 6666 7,0.0835 5034 7222 2222 2,
     $        0.1282 2657 4556 3271 6,0.2918 4902 6464 1404 6,
     $        0.8816 2726 7443 7576 5,3.3214 0828 1862 768,
     $        14.9957 6298 6862 6,78.9230 1301 1587,474.4515 3886 8,
     $        3207.4900 91,2 4086.5496,19 8923.12,179 1902.0,
     $        1748 4377.0)

      DATA(PI=3.1415 92653 58979)
      DATA(I=(0.0,1.0))
      DATA(ALPHA=0.853 667 218 838 951)
```

```
      CALL MAG ANGLE(Z,ZMAG,PHI)

      Z POWER = 1.0
      Z CUBED = Z**3

      IF(ZMAG .GT. 4.2) GO TO 50

      IF(ZMAG .LT. 3.2) 21,22
   21 N=12
      GO TO 30

   22 IF(ZMAG .LT. 4.1) 23,24
   23 N=15
      GO TO 30

   24 N=23

   30 SUM1 = 0.0
      SUM2 = 0.0
      SUM3 = 0.0
      SUM4 = 0.0

      DO 31 M=1,N
      TERM1 = A(M)*Z POWER
      TERM2 = B(M)*Z POWER
      TERM3 = C(M)*Z POWER
      TERM4 = D(M)*Z POWER
      SUM1 = SUM1+TERM1
      SUM2 = SUM2+TERM2
      SUM3 = SUM3+TERM3
      SUM4 = SUM4+TERM4
      Z POWER = -Z CUBED/200.0*Z POWER
   31 CONTINUE

      F = SUM1
      G = Z*SUM2
      GM2F = G-2.0*F
      F PRIME = -Z**02*SUM3
      G PRIME = SUM4
      GPMFP = G PRIME-2.0*F PRIME
      RT THIRD = SQRTF(1.0/3.0)

      H1 = G+I*RT THIRD*GM2F
      H1 PRIME = G PRIME+I*RT THIRD*GPMFP

      H2 = G-I*RT THIRD*GM2F
      H2 PRIME = G PRIME-I*RT THIRD*GPMFP

      RETURN


   50 SUM5 = 1.0
      SUM6 = 1.0
      SUM7 = 0.0
      SUM8 = 0.0
      Z REAL = Z
      Z IMAG = -I*Z
      I POWER = I
```

```
      M I POWER = -I
      RT Z = CPLX SQRT(Z)
      IF(Z IMAG .LT. 0.0) RT Z = -RT Z
      ONE RT Z = 1.0/RT Z
      Z FOURTH = CPLX SQRT(RT Z)
      REAL Z 4TH = Z FOURTH
      IF(REAL Z 4TH .LT. 0.0) Z FOURTH = -Z FOURTH
      BETA = ALPHA/Z FOURTH
      SQRT Z CUB = RT Z**3
      Z M3HALVS = 1.0/SQRT Z CUB
      Z M3HAFS M = Z M3HALVS
      EXP1 = CPLX EXPF(I*2.0/3.0*SQRT Z CUB)
      EXP2 = EXP1*CPLX EXPF(-I*5.0/12.0*PI)
      EXP3 = 1.0/EXP1*CPLX EXPF(I*5.0/12.0*PI)
      EXP4 = EXP1*CPLX EXPF(I*11.0/12.0*PI)
      EXP5 = 1.0/EXP1*CPLX EXPF(-I*11.0/12.0*PI)

      DO 51 M=1,14
      TERM5 = I POWER*CAP(M)*Z M3HAFS M
      TERM6 = M I POWER*CAP(M)*Z M3HAFS M
      SUM5 = SUM5+TERM5
      SUM6 = SUM6+TERM6
      EM = M
      SUM7 = SUM7+(-1.5*EM*1.0/Z)*TERM5
      SUM8 = SUM8+(-1.5*EM*1.0/Z)*TERM6
      Z M3HAFS M = Z M3HAFS M*Z M3HALVS
      I POWER = I POWER*I
      M I POWER = M I POWER*(-I)
   51 CONTINUE

      IF(Z REAL .GE. 0.0 .OR. Z IMAG .GE. 0.0) 61,62
   61 H1 = BETA*EXP2*SUM6
      H1 PRIME = BETA*EXP2*(SUM6*(-0.25*1.0/Z+I*RT Z)+SUM8)
      GO TO 70

   62 H1 = BETA*(EXP2*SUM6+EXP5*SUM5)
      H1 PRIME = BETA*(EXP2*(SUM6*(-0.25*1.0/Z+I*RT Z)+SUM8)+EXP5*(SUM5
     $           *(-0.25*1.0/Z-I*RT Z)+SUM7))

   70 IF(Z REAL .GE. 0.0 .OR. Z IMAG .LT. 0.0) 71,72
   71 H2 = BETA*EXP3*SUM5
      H2 PRIME = BETA*EXP3*(SUM5*(-0.25*1.0/Z-I*RT Z)+SUM7)
      RETURN

   72 H2 = BETA*(EXP3*SUM5+EXP4*SUM6)
      H2 PRIME = BETA*(EXP3*(SUM5*(-0.25*1.0/Z-I*RT Z)+SUM7)+EXP4*(SUM6
     $           *(-0.25*1.0/Z+I*RT Z)+SUM8))
      RETURN
      END
```

```
      SUBROUTINE R POLYNOM

      COMMON/INPUT/THETA,IN OMIT(98)
      COMMON/WG INPUT/WGIN SKIP(8),T LIST(11),WGIN OMIT(70)
      COMMON/R/LOG R(4),R SKIP(10),R(4)
      COMMON/INIT R/ADJ FLAG
      COMPLEX THETA,LOG R,R,LOG R MTRX,T LIST,LAGR INTP,CPLX EXPF
      INTEGER DIMENSN,ADJ FLAG,DEBUG
      DIMENSION LOG R MTRX(10,4)


      ENTRY GEN R POLY
      PRINT 100
  100 FORMAT(1H ,/////,29HR MATRIX INTERPOLATION POINTS)

      M = 1
   11 THETA = T LIST(M)
      CALL INTEG
      DO 12 N=1,4
   12 LOG R MTRX(M,N) = LOG R(N)
      IF(T LIST(M+1) .EQ. 0.0) GO TO 13
      ADJ FLAG = 1
      M = M+1 $ GO TO 11
   13 DIMENSN = M
      ADJ FLAG = 0

      PRINT 300
  300 FORMAT(1H0,6X,6HT LIST,14X,5H11R11,17X,3H1R1,
     $           17X,4H1R11,19X,4H11R1)
      DO 31 M=1,DIMENSN
   31 PRINT 301,T LIST(M),(LOG R MTRX(M,N),N=1,4)
  301 FORMAT(1H ,1X,C(F6.2,F6.2),4(2X,C(F10.4,F10.2)))
      RETURN


      ENTRY USE POLY

      DO 51 N=1,4
      LOG R(N) = LAGR INTP(T LIST,LOG R MTRX(1,N),THETA,DIMENSN)
   51 R(N) = CPLX EXPF(LOG R(N))
      RETURN

      END
```

```
      FUNCTION LAGR INTP(X,Y,X0,N)

      COMPLEX X,Y,X0,LAGR INTP,SUM,PROD
      DIMENSION X(N),Y(N)

      SUM = 0.0
      DO 12 J=1,N
      PROD = 1.0
      DO 11 I=1,N
      IF(I .EQ. J) GO TO 11
      PROD = PROD*(X0-X(I))/(X(J)-X(I))
   11 CONTINUE
   12 SUM = SUM+PROD*Y(J)

      LAGR INTP = SUM

      END
```

```
      SUBROUTINE INTEG
C BUDDEN

      COMMON/INPUT/IN SKIP(8),D,REL PREC,TOP HT,IN OMIT(89)
      COMMON/FLG INPUT/DEBUG,RC PRINT,FLG OMIT(23)
      COMMON/R/LOG R(8),DLOGRDH(8),HT,DELH,R OMIT(8)
      COMMON/1/L,HT LIST(256),OMIT1(256,3)
      REAL LOG R,LOG N LIST,LOGR 0,LOWEST HT,LOGE10
      INTEGER S FLAG,DEBUG,RC PRINT
      DIMENSION LOGR 0(8),DLOGRDH 0(8),
     $          DEL LOGR 0(8),DEL LOGR 1(8),DEL LOGR 2(8)
      DATA(DELHMIN=0.03125)
      DATA(LOGE10=2.302585093)


      FACTOR = EXPF(-REL PREC*LOGE10)
      EMAX = 3.0*FACTOR
      EMIN = 0.3*FACTOR
      L = 1
      HT = TOP HT
      DELH = DELHMIN
      SAVE DELH = DELH
      CALL INIT COMP
      CALL S MATRIX
      CALL INITIAL R
      CALL R DERIV
      IF(RC PRINT .EQ. 1 .OR. DEBUG .GE. 1) PRINT 100
  100 FORMAT(1H0,15X,21H R-MATRIX INTEGRATION )
      IF(RC PRINT .EQ. 1 .OR. DEBUG .GE. 1) CALL R COL HEAD
      IF(RC PRINT .EQ. 1 .OR. DEBUG .GE. 1) CALL PRINT R

C RUNGE KUTTA
   20 S FLAG = 0
      IF(HT .EQ. HT LIST(L+1)) L = L+1
      IF(HT-DELH .GE. HT LIST(L+1)) GO TO 21
      DELH = HT-HT LIST(L+1)
      S FLAG = 1
   21 IF(HT-DELH .LT. D) DELH = HT-D

      DO 31 I=1,8
      LOGR 0(I) = LOG R(I)
   31 DLOGRDH 0(I) = DLOGRDH(I)
C TRY AGAIN
   33 DO 34 I=1,8
      DEL LOGR 0(I) = -DLOGRDH 0(I)*DELH
   34 LOG R(I) = LOGR 0(I)+0.5*DEL LOGR 0(I)

      HT = HT-0.5*DELH
      CALL S MATRIX
      CALL R DERIV

      DO 35 I=1,8
      DEL LOGR 1(I) = -DLOGRDH(I)*DELH
   35 LOG R(I) = LOGR 0(I)+0.5*DEL LOGR 1(I)

      CALL R DERIV
```

```
      DO 36 I=1,8
      DEL LOGR 2(I) = -DLOGRDH(I)*DELH
   36 LOG R(I) = LOGR 0(I)+DEL LOGR 2(I)

      HT = HT-DELH+0.5*DELH
      CALL S MATRIX
      CALL R DERIV

      ERROR = 0.0
      DO 37 I=1,8
      DEL LOGR 4 = ((-DLOGRDH(I)*DELH+DEL LOGR 0(I))/2.0
     $           +DEL LOGR 1(I)+DEL LOGR 2(I))/3.0
      LOG R(I) = LOGR 0(I)+DEL LOGR 4
      ERR = ABSF(DEL LOGR 2(I)-DEL LOGR 4)
      IF(ERROR .LT. ERR) ERROR = ERR
   37 CONTINUE
      IF(ERROR .GT. EMAX .AND. DELH .GT. DELHMIN) 38,40
   38 HT = HT+DELH
      DELH = DELH/2.0
      IF(DELH .LT. DELHMIN) DELH = DELHMIN
      GO TO 33

   40 CALL R DERIV
      IF(RC PRINT .EQ. 1 .OR. DEBUG .GE. 1) CALL PRINT R
      IF(ERROR .LT. EMIN) DELH = 2.0*DELH
      IF(S FLAG .EQ. 1) DELH = SAVE DELH
      SAVE DELH = DELH
      IF(HT .GT. D) GO TO 20

      IF(RC PRINT .EQ. 1) CALL RATIO DIF
      RETURN

      END
```

```
 SUBROUTINE DERIV EQU

 COMMON/INPUT/THETA,FREQ KHZ,AZIMUTH,DIP ANGLE,MAG FIELD,ALPHA,H,
$           IN OMIT(92)
 COMMON/SP INPUT/NR SPEC,COEFF NU(3),EXP NU(3),CHARGE(3),M RATIO(3)
 COMMON/FLG INPUT/FLG SKIP(5),NEG ELECT,FLG OMIT(19)
 COMMON/R/LOG R11,LOG R22,LOG R12,LOG R21,
$           DLNR11DH,DLNR22DH,DLNR12DH,DLNR21DH,
$           HT,R SKIP,R11,R22,R12,R21
 COMMON/M/M11,M12,M13,M21,M22,M23,M31,M32,M33,C,S,CSQ
 COMMON/1/L,HT LIST(256),LOG N LIST(256,3)
 COMPLEX I,THETA,K OVER 2I,
$           C,S,CSQ,ONE OVER C,S OVER C,
$           ILY,IMY,INY,U,USQ,D,
$           M11,M12,M13,M21,M22,M23,M31,M32,M33,
$           T11,T31,T42,T44,
$           T12 OVER C,T14 OVER C,T32 OVER C,T34 OVER C,CT41,
$           S11A,D11A,S11B,D11B,S12,D12,S21,D21,S22,D22,
$           B11,B22,B12,B21,R12R21,C12,C21,
$           DERIV11,DERIV22,DERIV12,DERIV21,
$           R11,R22,R12,R21,LOG R11,LOG R22,LOG R12,LOG R21,
$           DLNR11DH,DLNR22DH,DLNR12DH,DLNR21DH,
$           CPLX LOGF,CPLX EXPF,CPLX COSF,CPLX SINF
 REAL MAG FIELD,M RATIO,LOG N,LOG N LIST,
$           LSQYSQ,MSQYSQ,NSQYSQ,LMYSQ,LNYSQ,MNYSQ
 DIMENSION X(5),Y(5),YSQ(5),Z(5),COLL FREQ(5),EN(5),
$           ILY(5),IMY(5),INY(5),LSQYSQ(5),MSQYSQ(5),NSQYSQ(5),
$           LMYSQ(5),LNYSQ(5),MNYSQ(5)
 DATA(PI=3.141592653)
 DATA(DEG TO RAD=0.01745329252)
 DATA(COEFF Y=1.758796E11),(COEFF X=3.182357E03)
 DATA(VEL LIGHT=2.997928E05)
 DATA(I=(0.0,1.0))


 ENTRY INIT COMP
 OMEGA = 2.0*PI*FREQ KHZ*1000.0
 K OVER 2I = (OMEGA/VEL LIGHT)/(2.0*I)
 SIN DIP = SINF(DIP ANGLE*DEG TO RAD)
 DIR COS L = SIN DIP*COSF(AZIMUTH*DEG TO RAD)
 DIR COS M = SIN DIP*SINF(AZIMUTH*DEG TO RAD)
 DIR COS N = -COSF(DIP ANGLE*DEG TO RAD)
 C = CPLX COSF(THETA*DEG TO RAD)
 CSQ = C*C
 S = CPLX SINF(THETA*DEG TO RAD)
 ONE OVER C = 1.0/C
 S OVER C = S/C

 DO 11 K=1,NR SPEC
 Y(K) = COEFF Y*CHARGE(K)*MAG FIELD/(OMEGA*M RATIO(K))
 YSQ(K) = Y(K)**2
 ILY(K) = I*DIR COS L*Y(K)
 IMY(K) = I*DIR COS M*Y(K)
 INY(K) = I*DIR COS N*Y(K)
 LSQYSQ(K) = DIR COS L**2*YSQ(K)
 MSQYSQ(K) = DIR COS M**2*YSQ(K)
 NSQYSQ(K) = DIR COS N**2*YSQ(K)
 LMYSQ(K) = DIR COS L*DIR COS M*YSQ(K)
```

```
   LNYSQ(K) = DI       L*DIR COS N*YSQ(K)
11 MNYSQ(K) = DI       M*DIR COS N*YSQ(K)
   RETURN

   ENTRY S MATRIX
   M11 = 0.0 $ M12 = 0.0 $ M13 = 0.0
   M21 = 0.0 $ M22 = 0.0 $ M23 = 0.0
   M31 = 0.0 $ M32 = 0.0 $ M33 = 0.0

   DO 24 K=1,NR SPEC
   LOG N = LOG N LIST(L+1,K)+(HT-HT LIST(L+1))
  $           /(HT LIST(L)-HT LIST(L+1))
  $           *(LOG N LIST(L,K)-LOG N LIST(L+1,K))
   EN(K) = EXPF(LOG N)
   X(K) = COEFF X*(1.0E06*EN(K))*CHARGE'K)**2/(OMEGA**2*M RATIO(K))
   IF(NEG ELECT .EQ. 1) X(K) = -X(K)
   COLL FREQ(K) = COEFF NU(K)*EXPF(EXP NU(K)*1000.0*HT)
   Z(K) = COLL FREQ(K)/OMEGA
   IF(NEG ELECT .EQ. 1) Z(K) = -Z(K)
   U = 1.0-I*Z(K)
   USQ = U*U
   D = -X(K)/(U*(USQ-YSQ(K)))
   M11 = M11+(USQ-LSQYSQ(K))*D
   M12 = M12+(-INY(K)*U-LMYSQ(K))*D
   M13 = M13+(IMY(K)*U-LNYSQ(K))*D
   M21 = M21+(INY(K)*U-LMYSQ(K))*D
   M22 = M22+(USQ-MSQYSQ(K))*D
   M23 = M23+(-ILY(K)*U-MNYSQ(K))*D
   M31 = M31+(-IMY(K)*U-LNYSQ(K))*D
   M32 = M32+(ILY(K)*U-MNYSQ(K))*D
24 M33 = M33+(USQ-NSQYSQ(K))*D

   CURV TERM = ALPHA*(H-HT)
   M11 = M11-CURV TERM
   M22 = M22-CURV TERM
   M33 = M33-CURV TERM


   D = 1.0/(1.0+M33)
   T11 = -S*M31*D
   T12 OVER C = S OVER C*M32*D
   T14 OVER C = (C+M33*ONE OVER C)*D
   T31 = M23*M31*D-M21
   T32 OVER C = C+(M22-M23*M32*D)*ONE OVER C
   T34 OVER C = S OVER C*M23*D
   CT41 = (1.0+M11-M13*M31*D)*C
   T42 = M32*M13*D-M12
   T44 = -S*M13*D

   S11A = T11+T44
   D11A = T11-T44
   S11B = T14 OVER C+CT41
   D11B = T14 OVER C-CT41
   S12 = T12 OVER C+T42
   D12 = T12 OVER C-T42
   S21 = T34 OVER C+T31
   D21 = T34 OVER C-T31
   S22 = C+T32 OVER C
   D22 = C-T32 OVER C
```

```
RETURN


ENTRY R DERIV
R11 = CPLX EXPF(LOG R11)
R22 = CPLX EXPF(LOG R22)
R12 = CPLX EXPF(LOG R12)
R21 = CPLX EXPF(LOG R21)

RL LOG R12 = LOG R12
IF(RL LOG R12 .LT. -11.0) R12 = 0.0
RL LOG R21 = LOG R21
IF(RL LOG R21 .LT. -11.0) R21 = 0.0

B11 = R11*(D11A-D11B)
B22 = R22*D22
B12 = R12*D21
B21 = R21*S12
R12R21 = R12*R21
C12 = R12*S21
C21 = R21*D12

DERIV11 = B11+B12+B21-S11B-S11B+(R12R21*D22+C12+C21-D11A-D11B)/R11
DERIV22 = B12+B21+B22-S22-S22+(R12R21*(D11A-D11B)+B12+B21+D22)/R22
DERIV12 = B11+B12+B22+S11A-S11B-S22+(R11*S12+D12)*(R22+1.0)/R12
DERIV21 = B11+B21+B22-S11A-S11B-S22+(R11*D21+S21)*(R22+1.0)/R21

DLNR11DH = K OVER 2I*DERIV11
DLNR22DH = K OVER 2I*DERIV22
DLNR12DH = K OVER 2I*DERIV12
DLNR21DH = K OVER 2I*DERIV21
RETURN

END
```

```
      SUBROUTINE INITIAL R

      COMMON/INPUT/IN SKIP(5),MAG FIELD,IN OMIT(94)
      COMMON/FLG INPUT/DEBUG,FLG OMIT(24)
      COMMON/R/LOG R11,LOG R22,LOG R12,LOG R21,R SKIP(10),
     $           R11,R22,R12,R21
      COMMON/INIT R/ADJ FLAG
      COMMON/M/M11,M12,M13,M21,M22,M23,M31,M32,M33,C,S,CSQ
      COMPLEX M11,M12,M13,M21,M22,M23,M31,M32,M33,C,S,CSQ,
     $           B3,B2,B1,B0,Q,C TEMP,
     $           D11,D13,D31,D33,DELTA,P,T,Q1,Q2,P1,P2,T1,T2,
     $           R11,R22,R12,R21,
     $           LOG R11,LOG R22,LOG R12,LOG R21,
     $           FN SQ,F ROOT,
     $           CPLX SQRT,CPLX LOGF,CONJ
      REAL MAG FIELD,MAG Q
      INTEGER DEBUG,ADJ FLAG
      DIMENSION DIFF(4),Q(4),P(2),T(2)
      DIMENSION PHASE R(8),PREV PHAS(8)
      DATA(PI=3.141592653)
      DATA(ADJ FLAG=0)
      EQUIVALENCE(Q(1),Q1),(Q(2),Q2),(P(1),P1),(P(2),P2),
     $           (T(1),T1),(T(2),T2)
      EQUIVALENCE(LOG R11,PHASE R)


      IF(MAG FIELD .EQ. 0.0) GO TO 50

      B3 = S*(M13+M31)/(4.0*(1.0+M33))
      B2 = (-(CSQ+M33)*(1.0+M11)+M13*M31-(1.0+M33)*(CSQ+M22)+M23*M32)
     $           /(6.0*(1.0+M33))
      B1 = S*(M12*M23+M21*M32-(CSQ+M22)*(M13+M31))/(4.0*(1.0+M33))
      B0 =((1.0+M11)*(CSQ+M22)*(CSQ+M33)+M12*M23*M31+M13*M21*M32
     $           -M13*(CSQ+M22)*M31-(1.0+M11)*M23*M32-M12*M21*(CSQ+M33))
     $           /(1.0+M33)

      CALL QUARTIC(B3,B2,B1,B0,Q,DEBUG)

      DO 12 N=1,4
      CALL MAG ANGLE(Q(N),MAG Q,ANGLE Q)
      IF(ANGLE Q .LT. 135.0) ANGLE Q = ANGLE Q+360.0
12    DIFF(N) = ABSF(ANGLE Q-315.0)

      DO 15 M=2,4
      DO 15 N=M,4
      IF(DIFF(N) .GT. DIFF(M-1)) GO TO 15
      TEMP = DIFF(N) $ DIFF(N) = DIFF(M-1) $ DIFF(M-1) = TEMP
      C TEMP = Q(N) $ Q(N) = Q(M-1) $ Q(M-1) = C TEMP
15    CONTINUE

      DO 21 N=1,2
      D11 = 1.0+M11-Q(N)**2
      D13 = M13+S*Q(N)
      D31 = M31+S*Q(N)
      D33 = 1.0+M33-S**2

      DELTA = D11*D33-D13*D31
      P(N) = (-M12*D33+D13*M32)/DELTA
```

```
      T(N) = Q(N)*P(N)-S*(-D11*M32+M12*D31)/DELTA
      POYNTING = CONJ(P(N))*(Q(N)*P(N)-S*T(N))+Q(N)
      IF(POYNTING .LT. 0.0) PRINT 201,Q(N),POYNTING
  201 FORMAT(1H0,8HFOR Q = ,C(E11.3,E11.3),11HPOYNTING = ,E11.3)
   21 CONTINUE

      DELTA = (T1*C+P1)*(C+Q2)-(T2*C+P2)*(C+Q1)
      R11 = ((T1*C-P1)*(C+Q2)-(T2*C-P2)*(C+Q1))/DELTA
      R22 = ((T1*C+P1)*(C-Q2)-(T2*C+P2)*(C-Q1))/DELTA
      R12 =-2.0*C*(T1*P2-T2*P1)/DELTA
      R21 =-2.0*C*(Q1-Q2)/DELTA

   22 LOG R11 = CPLX LOGF(R11)
      LOG R22 = CPLX LOGF(R22)
      LOG R12 = CPLX LOGF(R12)
      LOG R21 = CPLX LOGF(R21)

      IF(ADJ FLAG .EQ. 0) GO TO 36
      DO 35 N=2,8,2
   33 IF(PHASE R(N)-PREV PHAS(N) .LE. PI) GO TO 34
      PHASE R(N) = PHASE R(N)-2.0*PI $ GO TO 33
   34 IF(PREV PHAS(N)-PHASE R(N) .LE. PI) GO TO 35
      PHASE R(N) = PHASE R(N)+2.0*PI $ GO TO 34
   35 CONTINUE
   36 DO 37 N=2,8,2
   37 PREV PHAS(N) = PHASE R(N)
      RETURN

C RS BY FRESNEL
   50 FN SQ = 1.0+M11
      F ROOT = -CPLX SQRT(FN SQ-S*S)
      R11 = (FN SQ*C-F ROOT)/(FN SQ*C+F ROOT)
      R22 = (C-F ROOT)/(C+F ROOT)
      R12 = 0.0 $ R21 = 0.0
      GO TO 22

      END
```

```
      SUBROUTINE R OUTPUT

      COMMON/R/LOG R11,R11 ANGLE,LOG R22,R22 ANGLE,
     $         LOG R12,R12 ANGLE,LOG R21,R21 ANGLE,
     $         R SKIP(8),HT,R OMIT(9)
      REAL LOG R11,LOG R22,LOG R12,LOG R21
      DATA(RAD TO DEG=57.29577951)


      ENTRY R COL HEAD
      PRINT 100
  100 FORMAT(1H0,8X,2HHT,10X,5H11R11,17X,3H1R1,16X,4H1R11,16X,4H11R1)
      RETURN


      ENTRY PRINT R
      R11 MAG = EXPF(LOG R11)
      R22 MAG = EXPF(LOG R22)
      R12 MAG = EXPF(LOG R12)
      R21 MAG = EXPF(LOG R21)
      DEG11 = R11 ANGLE*RAD TO DEG
      DEG22 = R22 ANGLE*RAD TO DEG
      DEG12 = R12 ANGLE*RAD TO DEG
      DEG21 = R21 ANGLE*RAD TO DEG

      PRINT 200,HT,R11 MAG,DEG11,R22 MAG,DEG 22,R12 MAG,DEG12,
     $         R21 MAG,DEG21
  200 FORMAT(1H ,F10.2,4(F10.4,F10.2))
      RETURN


      ENTRY RATIO DIF
      RATIO = R12 MAG/R22 MAG
      DIFF = DEG12-DEG22

      PRINT 301,RATIO
  301 FORMAT(1H0,19H1R11 MAG/1R1 MAG = ,E12.5)
      PRINT 302,DIFF
  302 FORMAT(1H ,23H1R11 ANGLE-1R1 ANGLE = ,F11.5)
      PRINT 303
  303 FORMAT(1H1)
      RETURN

      END
```

REFERENCES

1. Smith, R., "Program for the full-wave calculation of reflection in an arbitrary ionosphere (U)," DASA Report 1615, 1965.

2. Gossard, E.E., et al, "A computer program for vlf mode constants in an earth-ionosphere waveguide of arbitrary electron density distribution," MIPR DASA No. 558-64, 1966.

3. Gossard, E.E., Y. Gough, V. Noonkester, R. Pappert, I. Rothmuller, R. Smith, and P. Winterbauer, "A computer program for vlf mode constants in an earth-ionosphere waveguide of arbitrary electron density distribution," Navy Electronics Laboratory Interim Report No. 671 for DASA, September 1966.

4. McKinley, R.R., "A rigorous waveguide program for a cylinderical waveguide," NEL Tech Memo 1315, Oct. 1967.

5. Pappert, R.A. and C.H. Sheddy, "An accelerated waveguide program," Navy Electronics Laboratory Tech Memo 1040, Jan 1967.

6. Pappert, R.A., "On the problem of horizontal dipole excitation of the earth-ionosphere waveguide," NEL Interim Report 684 for DASA, April 1968.

7. Sheddy, C.H., Y. Gough, and R.A. Pappert, "An improved computer program for vlf mode constants in an earth-ionosphere waveguide of arbitrary electron density distribution," NEL Interim Report No. 682 for DASA, February 1968.

8. Budden, K.G., "The numerical solution of differential equations governing reflection of long radio waves from the ionosphere," Proc. Roy. Soc. (London), A 227, 516 (1955).

9. Sheddy, C.H., "A general analytic solution for reflection from a sharply bounded anisotropic ionosphere," to be published in Radio Science, August 1968.

10. Budden, G.G., "The influence of the earth's magnetic field on radio propagation by waveguide modes," Proc. Ray. Soc. (London) Ser. A265, No. 1323, 538-553, (1962).

11. Pappert, R.A., E.E. Gossard, I. Rothmuller, "A numerical investigation of classical approximations used in vlf propagation," Radio Science, 2 No. 4, 387-400.

12. Staff of the Computation Laboratory at Cambridge, Mass., "Tables of the modified Hankel functions of order one-third and of their derivatives," (Harvard Univ. Press, Cambridge, Mass) (1945).

13. Wait, J.R., "Electromagnetic waves in stratified media," p. 221 (Pergamon Press, New York, N.Y.), (1962).

14. Pappert, R.A., "A numerical sutdy of vlf mode structure and polarization below an anisotropic ionosphere," Radio Science, 3, No. 3, 219-233, 1968.

INITIAL DISTRIBUTION LIST

Director
Defense Atomic Support Agency
Attn: RAAE
Washington, D. C. 20305

Director (2)
Defense Atomic Support Agency
Attn: Technical Library
Washington, D. C. 20305

Commander
Test Command
Defense Atomic Support Agency
Sandia Base, New Mexico 87115

Director of Defense Research & Engineering
Attn: Atomic Programs
Washington, D. C. 20301

Commander
Joint Task Force EIGHT
Sandia Base
Attn: Technical Library
Albuquerque, N. Mex. 87115

Atomic Energy Commission
Albuquerque Operations Office
Attn: Dr. D. W. Sherwood
P.O. Box 5400
Albuquerque, N. Mex. 87100

U.S. Atomic Energy Commission
Special Asst. for Disarmament
Attn: Dr. Allen M. Labowitz
Washington, D. C. 20545

Director of Military Application
U.S. Atomic Energy Commission
Attn: Mr. Donald Gale
Washington, D. C. 20545

Director
Defense Intelligence Agency
Attn: Hq-Tr; Mr. J. H. Stewart
Washington, D. C. 20301

Director Nuclear Test Detection
Advanced Research Projects Agency
Washington, D. C. 20301

Director (2)
Advanced Research Projects Agency
Attn: F. A. Koether
Dr. Paul Tamarkin
Washington, D. C. 20301

Director
Weapons Systems Evaluation Group
Attn: Library
Washington, D. C. 20301

Defense Communications agency (2)
Attn: Mr. Probst Code 513
Dep Director for NMCSTS
8th & So. Courthouse Road
Arlington, Va. 22204

ITSA/ESSA (6)
Attn: A. Glenn Jean
W. Taylor
J. R. Johler
R. Knecht
D. D. Crombie
L. A. Berry
Boulder, Colorado 80302

National Bureau of Standards (2)
U.S. Department of Comerce
Attn: Mr. Arthur Ernst
Mr. Raymond T. Moore
Washington, D. C. 20234

Defense Documentation Center (20)
Cameron Station
Alexandria, Va. 22314

Chairman
Joint Chiefs of Staff
Attn: J-6
Washington, D. C. 20301

Chief of Research & Development
Department of the Army
Attn: LTCOL Dayton
Washington, D. C. 20310

Director
Army Ballistic Research Laboratories
Attn: Technical Library
Aberdeen Proving Grounds, Md. 21005

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* Navy Electronics Laboratory Center San Diego, Calif 92152	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

A FORTRAN program for mode constants in an earth-ionosphere waveguide

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

Research Report

5. AUTHOR(S) *(First name, middle initial, last name)*

R. Pappert, C.H. Sheddy, Y. Gough, W. Moler

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
31 May 1968	58	14

8a. CONTRACT OR GRANT NO. DASA MIPR 517-68 b. PROJECT NO. DASA Subtask RHA 2042 c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) Interim Report No. 683 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

10. DISTRIBUTION STATEMENT

Distribution of the document is unlimited

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Defense Atomic Support Agency Washington, D.C.

13. ABSTRACT

An updated version of an earlier waveguide program has been written in the FORTRAN compiler language. The new program includes the following alterations or additions; (1) An improved Runge-Kutta routine; (2) A procedure which finds directly the starting solutions at the top of the ionosphere; (3) Provision for including up to five charged particles; (4) A method for economically finding approximate waveguide solutions; and modifications for use of the program for ELF wave propagation.

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6801

UNCLASSIFIED

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Waveguide Modes						
VLF Propagation						
Ionospheric Reflection						
FORTRAN Program						